МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Поркуян О. В., Тімошин А. С., Давиденко В. М.

ПРАКТИКУМ
З ДИСКРЕТНОЇ МАТЕМАТИКИ

2016

УДК 519.8
ББК 22.1

Рекомендовано Вченою Радою
Східноукраїнського національного університету імені Володимира Даля
як навчальний посібник для студентів спеціальності
«Комп'ютерні науки та інформаційні технології» вищих навчальних закладів
(протокол № 7 від 29.03.2016 р.)

Р е ц е н з е н т и:
Дрозд Ю. А. – доктор фізико-математичних наук, професор, член-кор. НАН України, завідувач відділу алгебри Інституту математики НАН України
Угрюмов М. Л. – доктор технічних наук, професор, професор кафедри інформатики Національного аерокосмічного університету ім. М. Є. Жуковського «Харківський авіаційний інститут»
Власенко К. В. – доктор педагогічних наук, професор кафедри вищої математики Донбаської державної машинобудівної академії

Поркуян О. В., Тімошин А. С., Давиденко В. М.
Практикум з дискретної математики: Навчальний посібник / О. В. Поркуян, А. С. Тімошин, В. М. Давиденко. – Харків: Видавництво та друкарня «Технологічний Центр», 2016. – 144 с.

ISBN 978-617-7319-03-9

Навчальний посібник містить основні теоретичні відомості курсу дискретної математики, приклади виконання завдань з кожної теми дисципліни та завдання для самостійного виконання.

УДК 519.8
ББК 22.1

ISBN 978-617-7319-03-9

ЗМІСТ

ПЕРЕДМОВА

Дисципліна «Дискретна математика» все далі набуває важливого значення у підготовці спеціалістів технічних спеціальностей, зокрема спеціалістів комп'ютерних наук та інформаційних технологій.

Зміст сучасної математичної підготовки спеціалістів технічного та комп'ютерного профілю все більше наповнюється задачами, пов'язаними з комбінаторикою, логікою, графами. Більш того, ці задачі мають алгоритмічний характер, і тому їх викладення потребує ретельного відношення.

Останнім часом у вищих навчальних закладах все більшу увагу приділяють організації самостійного вивчення дисциплін. Використання різних платформ дистанційного навчання потребує від викладача високого професійного та науково-методичного рівня. Від чіткості поставлених задач залежить якість їх виконання. Саме ця задача була поставлена авторами при підготовці цього практикуму.

Важливим аспектом побудови практикуму є реалізація компетентнісного підходу у навчанні дискретної математики, який базується на сукупності загальних принципів визначення цілей освіти, відбору змісту освіти, організації освітнього процесу та оцінки освітніх результатів.

Структура кожної окремої теми практикуму вибрана таким чином – це назва теми, ключові слова, основні теоретичні відомості, приклади виконання завдань, завдання для самостійного виконання. Особлива увага приділялася прикладам виконання завдань. Ця особливість помітно відрізняє практикум від звичайного збірнику задач.

Практикум містить 24 теми. Теми охоплюють теорію множин, бінарні відношення, булеві функції, оптимальні коди, комбінаторику та теорію графів. Розподіл матеріалу дискретної математики по темам обумовлено можливістю викласти певний метод або алгоритм та скласти чіткі завдання для самостійного виконання.

Дві перші теми стосуються теорії множин. Для доведення тотожностей використовуються графічний метод (діаграми Ейлера-Венна), аналітичні викладки на основі властивостей операцій над множинами, принцип включення та таблиці належності. Досить цікавим є питання досконалої нормальної форми Кантора (ДНФК) множини [9]. Приведено повну схему отримання ДНФК множини та її мінімізації.

При викладенні бінарних відношень (тема 3) особлива увага приділяється матричному та графічному зображенню відношень. Приведено характерні ознаки матриць та графів відношень відповідно до властивостей цих відношень.

Велика кількість прикладів пов'язана з відношенням еквівалентності, як одним з відомих та основних відношень в математиці.

Булеві функції викладено в темах 5 – 11. Основний практичний матеріал цих тем стосується елементарних булевих функцій від двох змінних, побудови таблиць істинності, доведення тотожностей, побудови досконалої нормальної форми булевої функції та мінімізації нормальних форм. Ретельно викладені метод Квайна та метод сполучення індексів мінімізації булевих функцій, побудова полінома Жегалкіна [6]. Описані класи булевих функцій та надано приклади перевірки булевих функцій на належність до того або іншого класу.

Теми 12 та 13 присвячені побудові оптимальних кодів. Викладені алгоритми Фано та Хаффмана для побудови оптимального коду та алгоритми побудови кодового дерева Фано та кодового дерева Хаффмана [6].

В темі 14 викладені елементи комбінаторики. Основа увага приділяється прикладам з застосуванням бінома Ньютона.

Елементи теорії графів викладені в темах 15 – 24. В основі практичного матеріалу цього розділу є матричне зображення графів, визначення метричних властивостей (ексцентриситет, радіус, діаметр, центр) графів, визначення ізольованих, тупикових, висячих вершин, петель, контурів в графах, побудова цикломатичної матриці та матриці розрізів, визначення стійкості, покриття та щільності графів, визначення максимального потоку через сіть, розфарбування графів [9]. Розглянуто алгоритмічні задачі на графах, а саме, пошук відстані між вершинами орграфу, знаходження довжини найкоротшого шляху між двома вершинами, побудова ейлеревого циклу в графах.

Наповнення змістом будь-якого навчального посібника в значній мірі залежить від авторів, їх переваг тих або інших тем, розділів. Змістові модулі дисципліни «Дискретна математика» досить стандартні, що можна прослідити при аналізі сучасних підручників, як вітчизняних так і зарубіжних [1], [2], [3], [4]. Отже, залишається тільки як можна доступніше та яскравіше на конкретний прикладах викласти те або інше питання курсу. Сподіваємося на те що нам це вдалося.

ТЕМА 1.

ДОВЕДЕННЯ ТОТОЖНОСТЕЙ В ТЕОРІЇ МНОЖИН.

Ключові слова: універсум, булеан, діаграма Ейлера-Венна, закон поглинання, теорема Де Моргана, закони склеювання, закони Порецького, принцип включення, таблиці належності.

1.1. Основні теоретичні відомості.

Поняття множини в математиці є первинним, тобто його означення має більш інтуїтивний характер, ніж строгий. Під множиною будемо розуміти сукупність однорідних елементів певної природи.

Множину *A* називають підмножиною множини *B* (пишуть: $A \subset B$), якщо кожний елемент множини *A* належить і множині *B*.

Якщо досліджувані множини є підмножинами якоїсь множини *U*, то цю множину *U* будемо називати *універсумом*. Множина всіх підмножин універсуму *U*, називається *булеаном* і позначається як $B(U)$.

Множину, яка не містить жодного елемента, називають *порожньою* та позначають $\varnothing$.

Якщо об’єкт *a* є елементом деякої множини *M*, то це позначають так: $a \in M$ (читається: *a* належить *M*, або *a* є елементом *M*). Той факт, що елемент *a* не належить множині *M*, позначають $a \notin M$.

Дві множини вважаються *тотожно рівними,* якщо вони складаються з тих самих елементів тобто $A = B \Leftrightarrow \forall x(x \in A \Leftrightarrow x \in B)$.

Найбільш розповсюджені способи зображення множин:

а) списком (перерахуванням) всіх елементів множини, записаних у фігурних дужках, наприклад, $\{x, y, z, \ldots\}$;

б) аналітичний (ідентифікатори множин зв'язані знаками операцій над множинами);

в) графічний (діаграми Ейлера-Венна, мограф, гіперкуб).

Розглянемо основні операції на множинах, такі як, доповнення, обєднання перетин, різниця, симетрична різниця.

Для означення цих операцій можна скористатись аналітичним або графічним способами.

Графічне зображення множин відбувається на *діаграмах Ейлера-Венна*. На цих діаграмах універсум *U* асоціюватимемо з внутрішністю прямокутника на площині, довільні множини – з кругами всередині прямокутника. Множина, як результат операції, буде виділятися на діаграмі більш темним кольором.

Доповненням до множини *A* (рис. 1.1) є множина, що містить всі елементи універсуму, за винятком елементів множини *A*.

Об’єднанням множин *A* і *B* (рис. 1.2) є множина, що складається з усіх елементів множини *A*, всіх елементів множини *B* і не містить жодних інших елементів.

Перетином множин *A* і *B* (рис. 1.3) є множина, що складається з тих і тільки тих елементів, які належать одночасно множині *A* і множині *B*.

Різницею множин *A* і *B* (рис. 1.4) є множина, що складається з тих і тільки тих елементів, які належать множині *A* й не належать *B*.

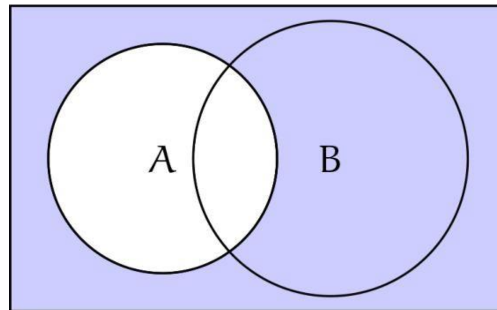


Рисунок 1.1 – Операція доповнення

$\overline{A} = \{x \in U \mid x \notin A\}$

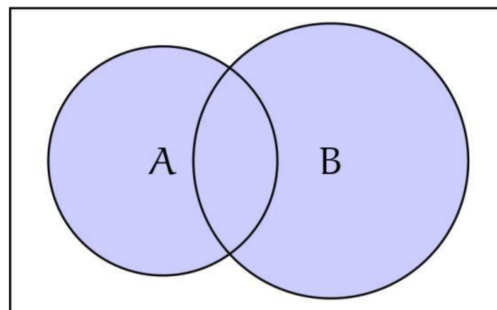


Рисунок 1.2 – Операція об’єднання

$A \cup B = \{x \in U \mid x \in A$ або $x \in B\}$

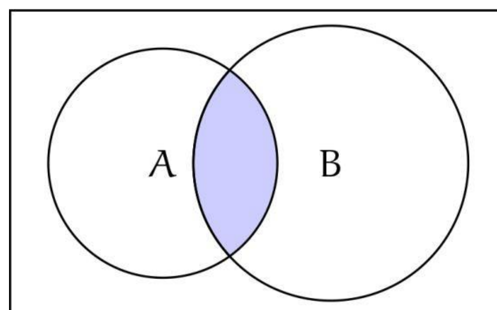


Рисунок 1.3 – Операція перетину

$A \cap B = \{x \in U \mid x \in A$ і $x \in B\}$

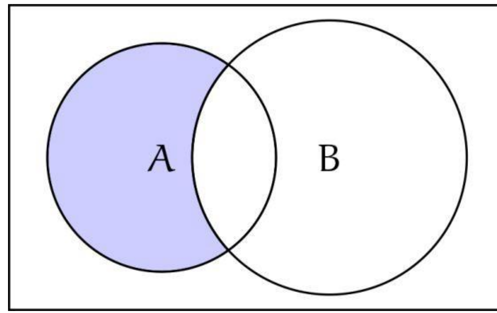


Рисунок 1.4 – Операція різниці

$A \square B = \{x \in U \mid x \in A \text{ і } x \notin B\}$

Симетричною різницею множин A і B (рис. 1.5) є множина, що складається з усіх елементів A, які не належать B, й усіх елементів B, які не належать A, та яка не містить ніяких інших елементів.

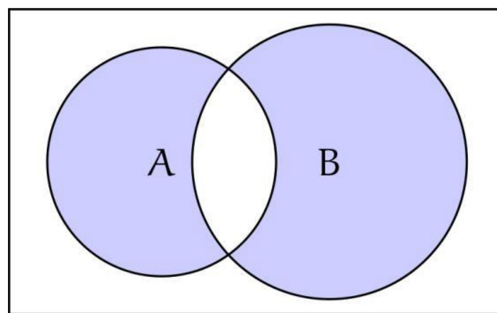


Рисунок 1.5 – Операція симетрична різниця

$A + B = \{x \in U \mid (x \in A \text{ і } x \notin B) \text{ або } (x \notin A \text{ і } x \in B\}$

Далі приведемо найважливіші властивості операцій, закони та тотожності, які пов'язують операції між собою.

Властивості універсуму U і порожньої множини $\varnothing$:

$$A \cup \varnothing = A \qquad A \cap U = A$$

$$A \cup \overline{A} = U \qquad A \cap \overline{A} = \varnothing$$

$$A \cup U = U \qquad A \cap \varnothing = \varnothing$$

$$\overline{\varnothing} = U \qquad \overline{U} = \varnothing$$

$$A + \varnothing = \varnothing + A = A$$

Закон самопоглинання (ідемпотентності) та поглинання:

$$A \cup A = A \qquad A \cap A = A$$

$$A \cup (A \cap B) = A \qquad A \cap (A \cup B) = A$$

Властивості комутативності, асоціативності та дистрибутивності:

$$A \cup B = B \cup A,\quad A \cap B = B \cap A,\quad A + B = B + A,$$

$$A \cup (B \cup C) = (A \cup B) \cup C,\quad A \cap (B \cap C) = (A \cap B) \cap C,$$

$$(A + B) + C = A + (B + C),$$

$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C),\quad A \cap (B \cup C) = (A \cap B) \cup (A \cap C).$$

Правила Де Моргана:

$$\overline{A \cup B} = \overline{A} \cap \overline{B},\quad \overline{A \cap B} = \overline{A} \cup \overline{B},$$

$$A - (B \cup C) = (A - B) \cap (A - C),\quad A - (B \cap C) = (A - B) \cup (A - C).$$

Правила склеювання: $(A \cap B) \cup (A \cap \overline{B}) = A,\ (A \cup B) \cap (A \cup \overline{B}) = A.$

Правила Порецького: $A \cup \overline{A} \cap B = A \cup B,\ A \cap (\overline{A} \cup B) = A \cap B.$

Інші корисні тотожності:

$$\overline{(\overline{A})} = A,\quad \overline{A} = U - A,\quad A - B = A \cap \overline{B},\quad A + B = (A \cap \overline{B}) \cup (\overline{A} \cap B).$$

При спрощенні аналітичних виразів з множинами чи приведенні їх до певного виду враховують наступний пріоритет операцій (у порядку зменшення): доповнення, перетин, об’єднання. Для зміни порядку виконання операцій використовуються дужки.

Всі властивості і закони, які приведено вище, як правило, доводяться на підставі *принципу включення*, який можна прослідити на прикладі 1 наступного пункту. Більш складні тотожності можуть бути доведені аналітичним методом, тобто спираючись на властивості і закони, або графічним методом, використовуючи діаграми Ейлера-Венна.

Крім того, доведення тотожностей може бути виконано за допомогою так званих *таблиць належності*, що містять всі можливі комбінації належності елементів множинам. Приведемо таблицю належності для основних операцій на множинах (0 вказує на те, що елемент не належить множині, 1 – на те, що належить):

A	B	$\overline{A}$	$A \cup B$	$A \cap B$	$A - B$	$A + B$
0	0	1	0	0	0	0
0	1	1	1	0	0	1
1	0	0	1	0	1	1
1	1	0	1	1	0	0

1.2. Приклади виконання завдань.

Приклад 1.1. Довести тотожність, використовуючи принцип включення

$$A-(B\cap C)=(A-B)\cup(A-C).$$

Припустимо, що $x\in A-(B\cap C)$. Тоді, $x\in A$ і $x\notin B\cap C$. Тобто, $x\in A$ і $x\notin B$ або $x\notin C$, або $x\in A$ і $x\notin B$ або $x\in A$ і $x\notin C$. А це означає, що $x\in(A-B)$ або $x\in(A-C)$. Остаточно, $x\in(A-B)\cup(A-C)$.

Тим самим доведено, що $A-(B\cap C)\subset(A-B)\cup(A-C)$.

Навпаки, нехай $x\in(A-B)\cup(A-C)$. Тоді $x\in(A-B)$ або $x\in(A-C)$. Тобто, $x\in A$ і $x\notin B$ або $x\in A$ і $x\notin C$. А це означає, що $x\in A$ і $x\notin B$ або $x\notin C$, або $x\in A$ і $x\notin B\cap C$.

Остаточно, $x\in A-(B\cap C)$. Отже, $(A-B)\cup(A-C)\subset A-(B\cap C)$.

Приклад 1.2. Довести тотожність за допомогою таблиць належності:

$$(A-(B\cup C))\cup(B-(A\cup C))\cup(C-(A\cup B))=A+B+C+(A\cap B\cap C).$$

Будемо використовувати таблиці належності для трьох множин. Користуючись таблицями належності основних операцій множин, послідовно заповнюємо стовпці для наступних виразів:

1.. $(B\cup C)$, 2. $(A\cup C)$, 3. $(A\cup B)$, 4. $A-(B\cup C)$, 5. $B-(A\cup C)$,
6. $C-(A\cup B)$, 7. $(A-(B\cup C))\cup(B-(A\cup C))\cup(C-(A\cup B))$,
8. $(A\cap B\cap C)$, 9. $A+B$, 10. $A+B+C$, 11. $A+B+C+(A\cap B\cap C)$.

A	*B*	*C*	1	2	3	4	5	6	**7**	8	9	10	**11**
0	0	0	0	0	0	0	0	0	**0**	0	0	0	**0**
0	0	1	1	1	0	0	0	1	**1**	0	0	1	**1**
0	1	0	1	0	1	0	1	0	**1**	0	1	1	**1**
0	1	1	1	1	1	0	0	0	**0**	0	1	0	**0**
1	0	0	0	1	1	1	0	0	**1**	0	1	1	**1**
1	0	1	1	1	1	0	0	0	**0**	0	1	0	**0**
1	1	0	1	1	1	0	0	0	**0**	0	0	0	**0**
1	1	1	1	1	1	0	0	0	**0**	1	0	1	**0**

Стовпець 7 та стовпець 9, які відповідають значенню лівої та правої частин тотожності, відповідно, однакові. Отже, тотожність доведена.

Приклад 1.3. Довести тотожність графічним методом:

$$(A-B)-C=(A-C)-(B-C).$$

Спочатку розглянемо ліву частину тотожності. Діаграми, які відповідають послідовно кожній операції зображено на рис. 1.6, рис. 1.7 та рис. 1.8. Далі, будуємо діаграми, які відповідають виразу правої частини тотожності (рис. 1.9, рис. 1.10, рис. 1.11). Діаграми на рис. 1.8 та рис. 1.11 співпадають, отже тотожність доведена.

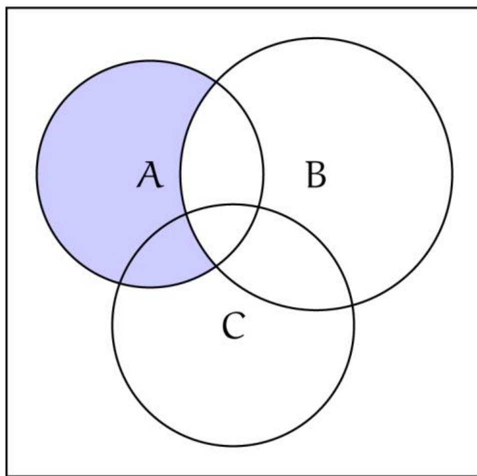


Рисунок 1.6 – Множина $A-B$

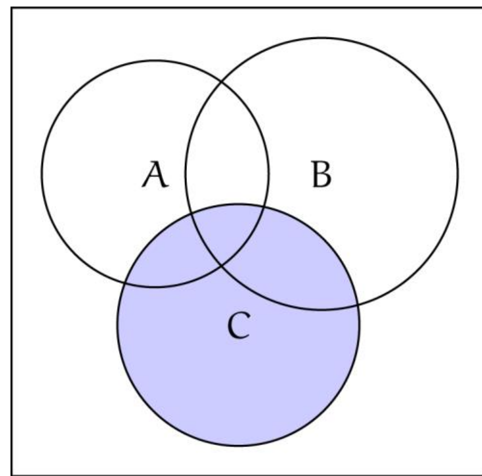


Рисунок 1.7 – Множина C

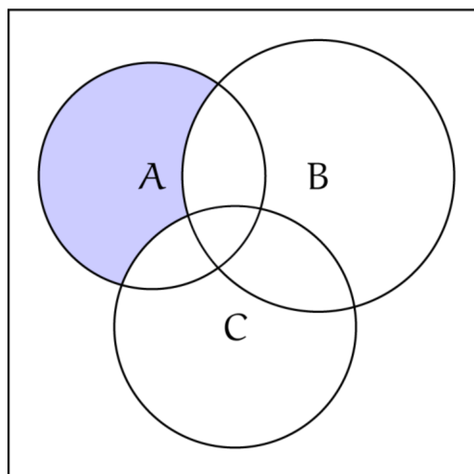


Рисунок 1.8 – Множина $(A-B)-C$

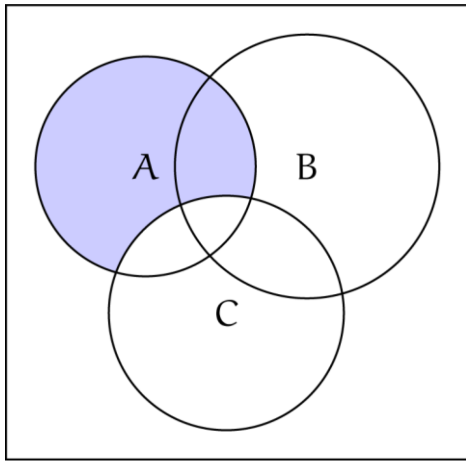


Рисунок 1.9 – Множина $A - C$

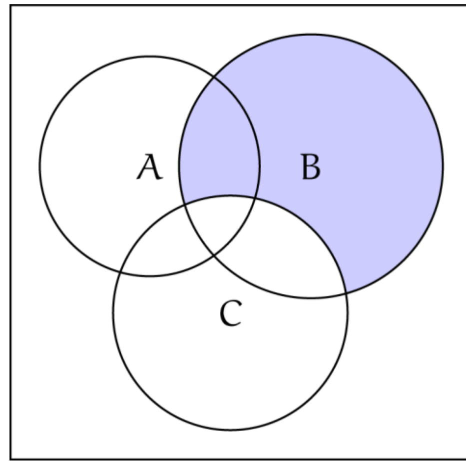


Рисунок 1.10 – Множина $B - C$

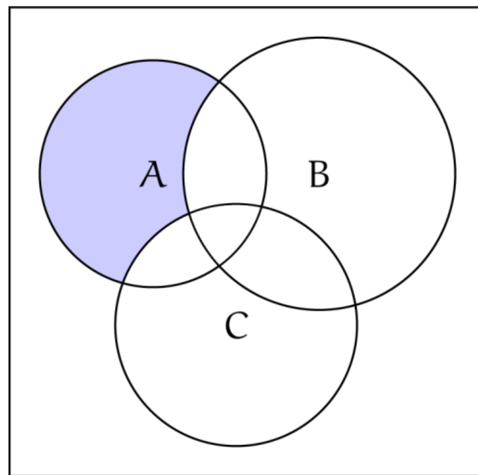


Рисунок 1.11 – Множина $(A - C) - (B - C)$

Приклад 1.4. Довести тотожність аналітичним методом:

$$(A - B) \cup (A \cap B) = A.$$

Для доказу цієї тотожності запишемо приведемо ліву частину тотожності до правої. Замінимо $A - B = A \cap \overline{B}$, тоді

$$(A - B) \cup (A \cap B) = (A \cap \overline{B}) \cup (A \cap B).$$

Згідно правилам склеювання:

$$(A - B) \cup (A \cap B) = (A \cap \overline{B}) \cup (A \cap B) = A \cap (\overline{B} \cup B).$$

Згідно властивостям універсаму та комутативності перетину, маємо

$$(A-B)\cup(A\cap B)=(A\cap\bar{B})\cup(A\cap B)=A\cap(\bar{B}\cup B)=A\cap U=A.$$

Приклад 1.5. Довести тотожність аналітичним методом:

$$A+B=(A-B)+(B-A).$$

Приведемо вираз $(A-B)+(B-A)$ до виразу $A+B.$

Використовуючи властивості різниці $A-B=A\cap\bar{B}$ та диз'юнктивної суми (симетричної різниці) $A+B=(A\cap\bar{B})\cup(\bar{A}\cap B)$ запишемо наступну послідовність рівних множин:

$$(A-B)+(B-A)=(A\cap\bar{B})+(B\cap\bar{A})=$$

$$=\left[(A\cap\bar{B})\cap\overline{(B\cap\bar{A})}\right]\cup\left[\overline{(A\cap\bar{B})}\cap(B\cap\bar{A})\right]=$$

/за законом Де Моргана $\overline{A\cap B}=\bar{A}\cup\bar{B}$/

$$=\left[A\cap\bar{B}\cap(\bar{B}\cup A)\right]\cup\left[(\bar{A}\cup B)\cap\bar{B}\cap A\right]=$$

/використовуючи закон поглинання $A\cap(A\cup B)=A$/

$$=(A\cap\bar{B})\cup(B\cap\bar{A})=A+B.$$

Приклад 1.6. Довести тотожність аналітичним методом:

$$(\bar{A}\cup B)+(\bar{B}\cup A)=(A\cap B)+(A\cup B).$$

Одним зі способів доведення тотожностей є приведення лівої та правої частин до одного виразу. Розглянемо спочатку ліву частину. Будемо використовувати вираз $A+B=(A\cap\bar{B})\cup(\bar{A}\cap B).$

$$(\bar{A}\cup B)+(\bar{B}\cup A)=\left((\bar{A}\cup B)\cap\overline{(\bar{B}\cup A)}\right)\cup\left((\bar{B}\cup A)\cap\overline{(\bar{A}\cup B)}\right)=$$

/далі використовуємо закон Де Моргана $\overline{A\cup B}=\bar{A}\cap\bar{B}$/

$$= \left(\left(\overline{A} \cup B\right) \cap B \cap \overline{A}\right) \cup \left(\left(\overline{B} \cup A\right) \cap A \cap \overline{B}\right) =$$

/враховуючи, що $A \cap (A \cup B) = A$, остаточно отримуємо/

$$= \left(\overline{A} \cap B\right) \cup \left(A \cap \overline{B}\right).$$

Розглянемо праву частину. Використовуючи вираз $A + B = (A \cap \overline{B}) \cup (\overline{A} \cap B)$, отримаємо

$$\left(A \cap B\right) + \left(A \cup B\right) = \left(A \cap B \cap \overline{\left(A \cup B\right)}\right) \cup \left(\overline{\left(A \cup B\right)} \cap \left(A \cap B\right)\right) =$$

/згідно законам Де Моргана/

$$= \left(A \cap B \cap \overline{A} \cap \overline{B}\right) \cup \left(\left(A \cup B\right) \cap \left(\overline{A} \cup \overline{B}\right)\right) =$$

/використовуємо властивість $A \cap \overline{A} = \varnothing$ та дистрибутивний закон/

$$= \left(\left(A \cup B\right) \cap \overline{A}\right) \cup \left(\left(A \cup B\right) \cap \overline{B}\right) = \left(\overline{A} \cap A\right) \cup \left(\overline{B} \cap A\right) \cup \left(\overline{A} \cap B\right) \cup \left(\overline{B} \cap B\right) =$$

$$= \left(B \cap \overline{A}\right) \cup \left(A \cap \overline{B}\right).$$

Результати перетворень лівої та правої частин тотожності співпали, то ж тотожність доведена.

1.3. Завдання для самостійного виконання.

Аналітичним та графічним способом (метод діаграм Ейлера-Венна) довести тотожності:

1. $\left(A - B\right) + \left(\left(A + C\right) \cap B\right) = \left(A - C\right) + \left(\left(A + B\right) \cap C\right).$
2. $\left(A + \left(A - B\right)\right) \cap \left(U - B\right) = \varnothing.$
3. $\left(A \cup \overline{C}\right) \cap \left(\overline{A} \cup \overline{B}\right) \cap \left(\overline{B} \cup \overline{C}\right) \cap \left(A \cup B\right) \cap \left(B \cup C\right) = \varnothing.$
4. $\left(A \cap B\right) \cup \left(B \cap C\right) \cup \left(A \cap C\right) = \left(A \cup B\right) \cap \left(B \cup C\right) \cap \left(A \cup C\right).$
5. $A + B = \left(\overline{A} \cup B\right) + \left(A + \overline{B}\right).$
6. $A - \left(B \cup C\right) = \left(A - B\right) \cap \left(A - C\right).$

7. $(C\cap B)\cup(B\cup\overline{A})\cup\overline{(B\cap C)}\cup(A\cap B)\cup(A\cap C)=U.$

8. $((\overline{B}\cap A)\cup C)\cap((A\cup\overline{C})\cup B)=B\cap C.$

9. $(A-B)\cup(B-C)\cup(C-A)=(A\cup B\cup C)+(A\cap B\cap C).$

10. $(A-B)\cup(B-A)=(B-A)+(A-B).$

11. $((A+B)-C)=(A-(B\cup C))\cup(B-(A\cup C)).$

12. $((A\cap C)+(B\cap D))=((A+B)\cup(C+D)).$

13. $(A\cap B\cap C)+(A\cup B\cup C)=(A\cup B\cup C)/(A\cap B\cap C).$

14. $(A\cup B)+(C\cup D)=A+B+C+D$, якщо $A\cap B=C\cap D$.

15. $(\overline{B}\cap C)\cup(A\cup\overline{B})\cup(\overline{A}\cap C)\cap((A\cap B)\cup(B\cap C))=\varnothing.$

16. $((A\cap\overline{B})\cup(B\cap C))\,\overline{\cap}\,((A\cap B)\cup(B\cap C))=\varnothing.$

17. $(A-B)+(B-A)=A+B.$

18. $(A\cap B)-(C\cup D)=(A-C)\cap(B-D).$

19. $((A\cup B)\cap C)\cup(\overline{A}\cap(\overline{B}\cup\overline{C}))=\overline{A}\cup C.$

20. $(A\cup B)+(A\cup C)+(B\cup C)=(A\cap B)+(A\cap C)+(B\cap C).$

21. $(A\cap B)\cup(C\cap D)=(A\cup C)\cap(B\cup C)\cap(A\cap D)\cap(B\cap D).$

22. $A-(B\cap C)=(A-B)\cup(A-C).$

23. $A\cap(B-C)=(A\cap B)-(A\cap C)=(A\cap B)-C.$

24. $(A\cap B)\cup(A\cap\overline{B})=(A\cup B)\cap(\overline{A}\cup B)=A.$

25. $A-(B-C)=(A-B)\cup(A\cap C).$

26. $A-(B\cup C)=(A-B)-C.$

27. $A\cup B=A-B-(A\cap B).$

28. $(((A\cup B)\cap(B\cup C))\cup(A\cup\overline{C})\cap(B\cup C)\cap(\overline{A}\cup B))=U.$

29. $(A\cup B)-C=(A-C)-(B\cup C).$

30. $(A-B)\cap C=(A\cap C)-(B\cap C).$

ТЕМА 2.

ДОСКОНАЛА НОРМАЛЬНА ФОРМА КАНТОРА МНОЖИНИ.

Ключові слова: терм, конституента, десятковий еквівалент, досконала нормальна форма Кантора (ДНФК), двійкова таблиця множини, мограф, таблиця Квайна, ядро покриття.

2.1. Основні теоретичні відомості.

Нехай $M_1, M_2, \ldots, M_n \in U,$ такі, що будь-яка множина $M \in U$ може бути породжена з цих множин за допомогою операцій доповнення, перетину та об'єднання.

Множину $M_i^{\sigma_i} = \begin{cases} M_i, \text{ при } \sigma_i = 1 \\ \overline{M}_i, \text{ при } \sigma_i = 0 \end{cases}$ називають *первинним термом*.

Множину виду

$$C = \bigcap_{i=1}^{n} M_i^{\sigma_i} = M_1^{\sigma_1} \cap M_2^{\sigma_2} \cap \ldots \cap M_n^{\sigma_n}$$

називають *конституентою*.

Конституенти мають наступні властивості:

а) переріз двох різних конституент є порожня множина;

б) об'єднання всіх конституент є універсум.

Кожній конституенті $C = \bigcap_{i=1}^{n} M_i^{\sigma_i}$ можна зіставити двійковий вектор $\sigma(C) = (\sigma_1 \sigma_2 \ldots \sigma_n)$ довжини n, в якому

$$\sigma_i = \begin{cases} 0, & \text{якщо } \overline{M_i} \in C; \\ 1, & \text{якщо } M_i \in C. \end{cases}$$

Очевидно, загальна кількість всіляких конституент дорівнює 2^n.

Крім того, зіставимо конституенті *десятковий еквівалент*:

$$d(C) = \sum_{i=1}^{n} \sigma_i 2^{i-1}.$$

Відомо, що будь-яка множина $M \in B(U)$ може бути задана у вигляді об'єднання конституент: $M = \bigcup_{j} \bigcap_{i=1}^{n} M_i^{\square_i}$. Цей вираз називають *досконалою нормальною формою Кантора* (ДНФК) множини *M*.

Взагалі, якщо множину задано об'єднанням перетинів попарно різних термів, то таке зображення множини називають нормальною формою Кантора (НФК). *Мінімальною* НФК множини М називається НФК цієї множини, яка має мінімальну складність. Під складністю зображення множини *M* будемо розуміти число *L*(*M*) символів M_i, $\overline{M_i}$, які складають нормальну форму.

Кожній множині $M \in B(U)$ може бути зіставлений двійковий вектор $\square(M)$ довжини 2^n, в якому *i*-тому розряду відповідає конституента з десятковим еквівалентом, який дорівнює *i*. Включення такої конституенти в множину *M* відповідає значенню 1 *i*-го розряду двійкового вектора, інакше – значенню 0.

Двійковому вектору $\square(M)$ відповідає число у десятковій системі числення, яке будемо називати *десятковим еквівалентом* множини *M* і позначати *d*(*M*).

При умові відомого двійкового вектора множини може бути побудована, так звана, *двійкова таблиця множини*, рядки якої містять двійкові вектори конституент, їх десяткові еквіваленти та відповідне значення розряду двійкового вектора множини. Отже, кількість рядків цієї таблиці дорівнює 2^n.

Наявність ДНФК множини дає можливість зображати множину графічно, а саме, на діаграмі Ейлера-Венна, у вигляді мографа та на гіперкубі.

Для випадку $n = 3$, на рис. 2.1, кожна область представлена десятковим еквівалентом відповідної конституенти.

Мограф множини є граф $G^M = \langle V, S_3 \rangle$, вершини якого є первинні терми універсуму $V = \{M_i, \overline{M}_i \mid i = 1, 2, 3\}$, а ребра відповідають наявним конституентам множини, точніше, $S_3 \subset V^3$. На рис. 2.2 зображено мограф універсуму, тобто $S_3 = V^3$. При відсутності певних конституент у ДНФК множини, відповідна трійка ребер на мографі теж відсутня.

Гіперкубом розмірності *n* називається граф, кожна вершина якого взаємно однозначно відповідає двійковому вектору простору розмірності *n*, а дві вершини поєднані ребром якщо їх двійкові вектори відрізняються тільки одним розрядом. На рис. 2.3 зображено гіперкуб розмірності 3. Зауважимо, що кожне

ребро гіперкуба теж має своє позначення. Наприклад, ребро, яке з’єднує вершини 011 та 001 позначається як 0-1.

На гіперкубі множина *M* позначається шляхом виділення вершин, відповідних конституентам, які складають ДНФК множини. Ребро гіперкубу, яке представлене певною парою суміжних виділених вершин, відповідає так званій *простій імпліканті*, яка утворюється в результаті склеювання відповідних вершинам конституент множини. Якщо в ДНФК множини замінити склеєні конституенти на відповідні прості імпліканти, одержимо *скорочену* НФК множини.

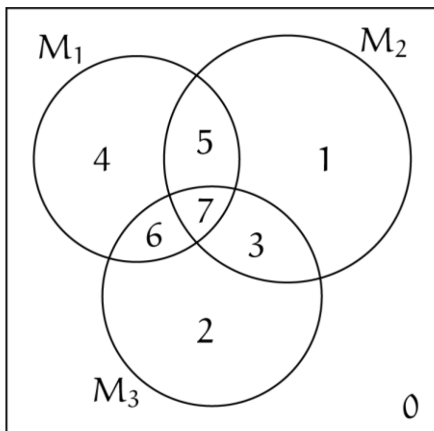


Рисунок 2.1

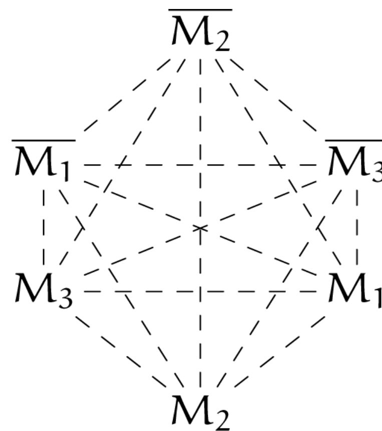


Рисунок 2.2

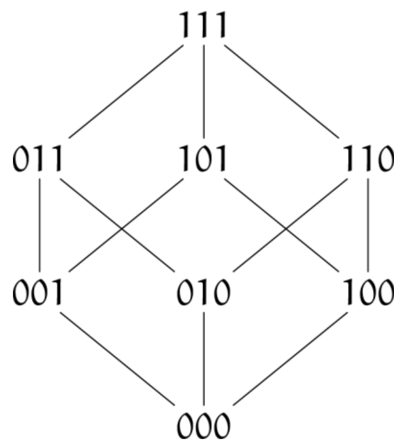


Рисунок 2.3

Мінімальна НФК множини *M* може бути отримана на основі скороченої НФК множини *M* за допомогою процедури покриття таблиці Квайна.

Таблиця Квайна є двовимірною таблицею з елементами 0 або 1, в якої рядкам відповідають прості імпліканти (виділені ребра гіперкуба), а стовпцям – вихідні конституенти (виділені вершини гіперкуба). На перетину *i*-го рядка та *j*-го стовпця стоїть одиниця, якщо *j*-та конституента входить до *i*-ої простої імпліканти, в іншому випадку ставимо 0.

Покриттям стовпців рядками двомірної таблиці називають таку множину рядків, коли для кожного стовпця знайдеться хоча б один рядок з цієї множини, на перетині з яким стовпець має одиницю, причому при видаленні хоча б одного рядка з цієї множини рядків вказана властивість не виконується. Якщо стовпець містить лише одну одиницю, то відповідний цій одиниці рядок називають *обов'язковим*.

Множина обов'язкових рядків утворює *ядро покриття*.

Отже, процедура покриття починається з визначення ядра покриття з наступним додаванням необхідних рядків.

Спрощені імпліканти, які відповідають рядкам, утворюючим покриття, входять до складу мінімальної НФК множини. На гіперкубі мінімальній НФК множини відповідає та мінімальна кількість ребер, яка покриває всі відповідні конституантам вершини.

2.2. Приклади виконання завдань.

Приклад 2.1. У просторі $U = \{M_1, M_2, M_3\}$ множина M задана десятковим еквівалентом $d(M) = 123$. Записати двійковий вектор множини, побудувати двійкову таблицю множини та її мограф, записати ДНФК множини, дати зображення множини на діаграмі Ейлера-Венна та гіперкубі. Мінімізувати ДНФК множини методом Квайна, вказати складність ДНФК та мінімізованого зображення множини.

Виконаємо переведення десяткового еквівалента множини до двійкової системи числення:

$123 = 0 \cdot 2^7 + 1\cdot 2^6 + 1\cdot 2^5 + 1\cdot 2^4 + 1\cdot 2^3 + 0 \cdot 2^2 + 1\cdot 2^1 + 1\cdot 2^0$.

Двійковий вектор множини: □(M) = (01111011).

Двійкова таблиця множини:

$d(C)$	M_1	M_2	M_3	M
0	0	0	0	1
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	1
5	1	0	1	1
6	1	1	0	1
7	1	1	1	0

ДНФК множини:

$$M = \overline{M}_1 \cap \overline{M}_2 \cap \overline{M}_3 \cup \overline{M}_1 \cap \overline{M}_2 \cap M_3 \cup \overline{M}_1 \cap M_2 \cap M_3 \cup$$
$$\cup M_1 \cap \overline{M}_2 \cap \overline{M}_3 \cup M_1 \cap \overline{M}_2 \cap M_3 \cup M_1 \cap M_2 \cap \overline{M}_3.$$

Множина M на діаграмі Ейлера-Венна (рис. 2.4) – це вся область за винятком областей 2 і 7. Мограф множини M вказано на рис. 2.5. Множина M зображена на гіперкубі (рис. 2.6) шляхом виділення вершин, які взяті у коло, та відповідних ребер, для яких виділені вершини є суміжними (позначені як суцільні).

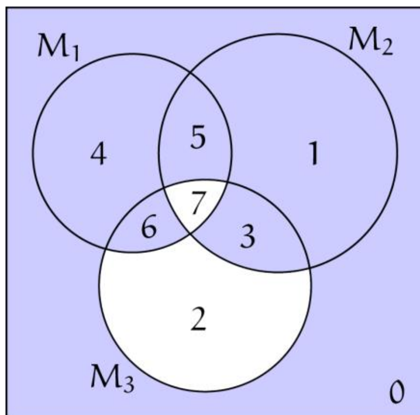


Рисунок 2.4

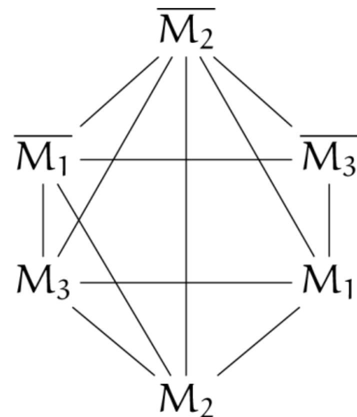


Рисунок 2.5

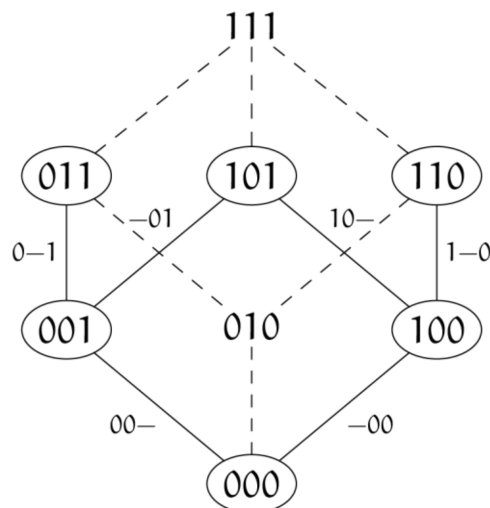


Рисунок 2.6

Таблиця Квайна:

	000	100	110	001	011	101
-00	1	1				
1-0		1	(1)			
00-	1			1		
0-1				1	(1)	
-01				1		1
10-		1				1

В даному випадку ядром покриття є рядки 1-0 та 0-1, які покривають 2-й 3-й, 4-й і 5-й стовпці. Для утворення покриття потрібно узяти 1-й і 5-й рядки (або 1-й і 6-й). В результаті отримуємо покриття {1-0, 0-1, -00, -01}, якому відповідає наступна мінімальна нормальна форма Кантора множини:

$$M(M_1, M_2, M_3) = \overline{M}_2 \cap \overline{M}_3 \cup M_1 \cap \overline{M}_3 \cup \overline{M}_1 \cap M_3 \cup \overline{M}_2 \cap M_3 .$$

В результаті спрощення складність $L(M)$ зменшилася від 18 до 8.

2.3. Завдання для самостійного виконання.

У просторі $U = \{M_1, M_2, M_3\}$ множина M задана десятковим еквівалентом $d(M)$ (табл. 2.1). Записати двійковий вектор множини, побудувати двійкову таблицю множини та її мограф, записати ДНФК множини, дати зображення множини на діаграмі Ейлера-Венна та на гіперкубі. Мінімізувати ДНФК множини методом Квайна, вказати складність ДНФК та мінімізованого зображення множини.

Таблиця 2.1

№	***d*(*M*)**	**№**	***d*(*M*)**	**№**	***d*(*M*)**	**№**	***d*(*M*)**
1	217	8	151	15	203	22	143
2	230	9	199	16	205	23	171
3	155	10	213	17	211	24	174
4	157	11	166	18	227	25	183
5	233	12	173	19	167	26	214
6	241	13	181	20	242	27	185
7	169	14	187	21	220	28	179

ТЕМА 3.

БІНАРНІ ВІДНОШЕННЯ ТА ЇХ ВЛАСТИВОСТІ.

Ключові слова: бінарне відношення, рефлексивне відношення, симетричне відношення, антисиметричне відношення, транзитивне відношення, матриця відношення, граф відношення.

3.1. Основні теоретичні відомості.

Нехай задано непорожню множину M. *Бінарне відношення* – це множина пар $(a, b) \in R \subseteq M \times M$, де $a, b \in M$. Використовують запис aRb, якщо $(a, b) \in R$, та $a \overline{R} b$, якщо $(a, b) \notin R$.

Бінарне відношення R на множині M називається *рефлексивним*, якщо $\forall a \in M$: aRa. Бінарне відношення R називається *антирефлексивним*, якщо $\forall a \in M$: $a \overline{R} a$.

Бінарне відношення R на множині M називається *симетричним* якщо $\forall a, b \in M$: $aRb \Rightarrow bRa$.

Антисиметричне відношення – це таке бінарне відношення, що $\forall a, b \in M$: $(a \square b \wedge aRb) \Rightarrow b\overline{R} a$, або інакше: $\forall a, b \in M$: $(aRb \wedge bRa) \Rightarrow a = b$.

Транзитивне відношення – це таке бінарне відношення, що $\forall a, b, c \in M$: $(aRb \wedge bRc) \Rightarrow aRc$.

При умові, якщо множина M скінченна, бінарне відношення R може бути зображене матрицею або орієнтованим графом. Бажано мати при цьому перелік всіх пар відношення R.

Матрицею відношення $R \subseteq M \times M$ будемо називати квадратну матрицю $[R]$ порядку $n \times n$, де $|M| = n$, елементами якої є 1 або 0, а саме, 1 – якщо пара (a,b) належить відношенню, 0 – в протилежному випадку. При цьому, рядки (стовпці) матриці пронумеровані елементами множини M.

Графом відношення будемо називати граф, точки якого відповідають елементам множин M, які з'єднуються орієнтованими дугами, якщо відповідні пари елементів належать відношенню.

Далі, приведемо деякі характерні ознаки для матриць та графів відношень з певними властивостями.

Матриця рефлексивного відношення, очевидно, має одиниці на головній діагоналі. Матриця симетричного відношення є симетричною.

Згідно означенню, для антисиметричного відношення має місце умова $[R] \cap [R]^T \subseteq E$, де перетин матриць знаходиться за правилом поелементного логічного добутку: $x_{ij} = \min(p_{ij}, q_{ij})$, де $p_{ij} \in [R]$, $q_{ij} \in [R]^T$, $x_{ij} \in [R] \cap [R]^T$.

Для матриці транзитивного відношення справедливе співвідношення $[R] \cdot [R] \subseteq [R]$ (добуток матриць виконується за правилами логічного добутку та логічної суми).

Граф рефлексивного відношення має петлі на кожній вершині. Граф симетричного відношення має подвійні ребра між вершинами.

3.2. Приклади виконання завдань.

Приклад 3.1. З'ясувати властивості для кожного з наступних бінарних відношень:

а) $\square = \{(1,2),(2,1),(1,1),(1,3),(3,2),(3,3)\}$ на множині $X = \{1,2,3\}$;

б) $\square = \{(x,y) : x - y \in Z\}$ на множині дійсних чисел R;

в) $\square = \{(x,y) : 2x = 3y\}$ на множині цілих чисел Z.

а) це відношення не є рефлексивним, бо для елемента $2 \in X$, $(2, 2) \notin \square$; не є симетричним, оскільки, наприклад, $(1,3) \in \square$, але $(3,1) \notin \square$; не є транзитивним, оскільки, наприклад $(3,2) \in \square$, $(2,1) \in \square$, але $(3,1) \notin \square$.

б) це відношення є рефлексивним, оскільки для довільного $x \in R$ різниця $x - x = 0 \in Z$; є симетричним, бо належність довільної пари (x, y) до відношення , означає $x - y = k \in Z$, але тоді $y - x = -k \in Z$, тобто пара $(y, x) \in \square$; є транзитивним, тому що для довільних $x, y, z \in Z$ належність пар (x, y) та (y, z) до відношення означає $x - y = k \in Z$ та $y - z = n \in Z$, але тоді $x - z = k + n \in Z$, тобто $(x, z) \in \square$.

в) це відношення не є рефлексивним, оскільки із всіх пар (x, x), $x \in Z$ тільки пара $(0,0) \in \square$, тому що для всіх інших $x \in Z$ не виконується рівність $2x = 3x$; не є симетричним, оскільки, наприклад, пара $(3, 2) \in \square$ $(2 \cdot 3 = 3 \cdot 2)$, а пара $(2,3) \notin \square$ $(2 \cdot 2 \neq 3 \cdot 3)$; не є транзитивним, оскільки, наприклад, $(9,6) \in \square$, $(6,4) \in \square$, але $(9, 4) \notin \square$.

Приклад 3.2. Визначити властивості відношення *R*, яке задано на множині $X = \{1,2,3,4,5\}$ матрицею:

$$[R] = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{pmatrix}.$$

а) відношення *R* рефлексивне, так як на головній діагоналі матриці [*R*] всі одиниці;

б) матриця [*R*] не симетрична, отже відношення *R* не є симетричним;

в) відношення *R* антисиметричне, оскільки:

$$[R] \cap [R]^T = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{pmatrix} \cap \begin{pmatrix} 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} =$$

$$= \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} = E\,.$$

г) для перевірки транзитивності знайдемо матрицю:

$$[R] \cdot [R] = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 \end{pmatrix}.$$

Оскільки $[R] \cdot [R] \not\subset [R]$, відношення *R* не є транзитивним.

Приклад 3.3. Бінарне відношення R на множині $U=\{1,2,3,4,5,6,7\}$ задане властивістю $R=\{(x,y): x+y=7\}$. Необхідно записати: всі пари відношення *R*, матрицю відношення *R* та побудувати граф відношення *R*.

Очевидно, $R=\{(1,6),(2,5),(3,4),(4,3),(5,2),(6,1)\}$.

Матриця відношення *R*:

$$[R]=\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Граф відношення *R* зображений на рисунку 3.1.

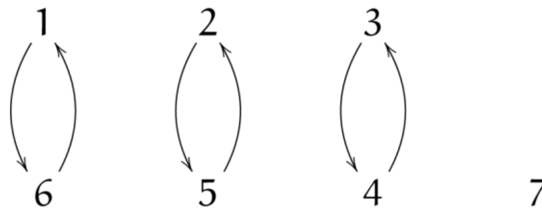


Рисунок 3.1

Приклад 3.4. Побудувати граф відношень заданих на множині $X=\{2,4,6,8\}$: а) R_1: «*x* менше *y*»; б) R_2: «*x* кратне *y*».

а) $R_1=\{(2,4),(2,6),(2,8),(4,6),(4,8),(6,8)\}$.

Граф відношення R_1: «*x* менше *y*» зображений на рис. 3.2.

б) $R_1=\{(2,2),(4,4),(6,6),(8,8),(4,2),(6,2),(8,2),(8,4)\}$.

Граф відношення R_2: «*x* кратне *y*» зображений на рис. 3.3.

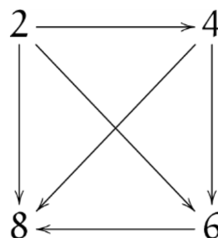


Рисунок 3.2 – Граф відношення «*x* менше *y*»

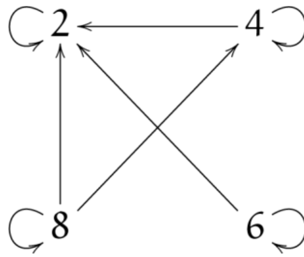


Рисунок 3.3 – Граф відношення «*x* кратне *y*»

Приклад 3.5. На множині відрізків (рис. 3.4) задано наступні відношення: *R*(⊥): «*x* перпендикулярний *y*», *R*(=): «*x* рівний *y*» та *R*(>): «*x* довший *y*». Побудувати графи та визначити властивості цих відношень.

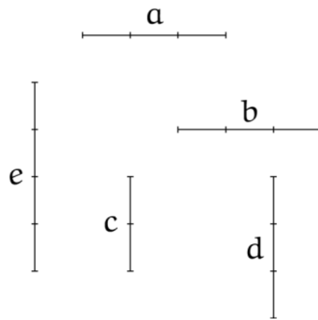


Рисунок 3.4

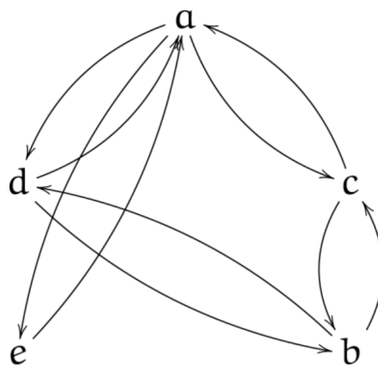


Рисунок 3.5 – Граф відношення *R*(⊥)

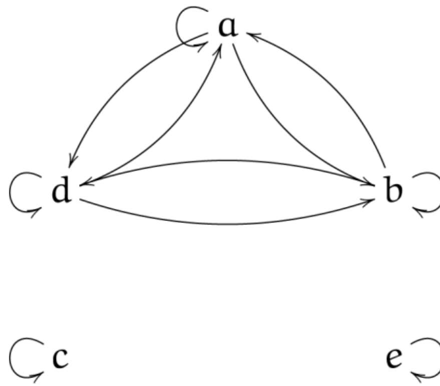


Рисунок 3.6 – Граф відношення *R*(=)

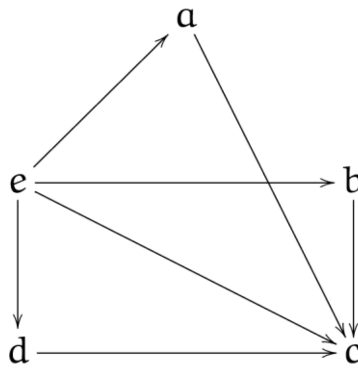


Рисунок 3.7 – Граф відношення *R*(>)

Відношення *R*(⊥) антирефлексивне, симетричне, не транзитивне.

Відношення *R*(=) рефлексивне, симетричне, транзитивне.

Відношення *R*(>) антирефлексивне, антисиметричне, транзитивне.

3.3. Завдання для самостійного виконання.

1. Дослідити вказані відношення на рефлексивність, симетричність та транзитивність:

а) «жити в одному місті»;

б) «бути знайомим»;

в) «бути молодшим»;

г) «бути нащадком»;

д) «бути дільником» на множині натуральних чисел;

е) «мати загальний спільний дільник, відмінний від одиниці» на множині натуральних чисел;

ж) «рівність» на множині дійсних чисел;

з) подібність трикутників на площині;

і) паралельність прямих на площині;

к) перпендикулярність прямих на площині.

2. З’ясуйте, який з графів, приведених на рис. 3.8, є графом відношення «x ділиться на y», заданого на множині $X = \{5, 10, 20, 30, 40\}$.

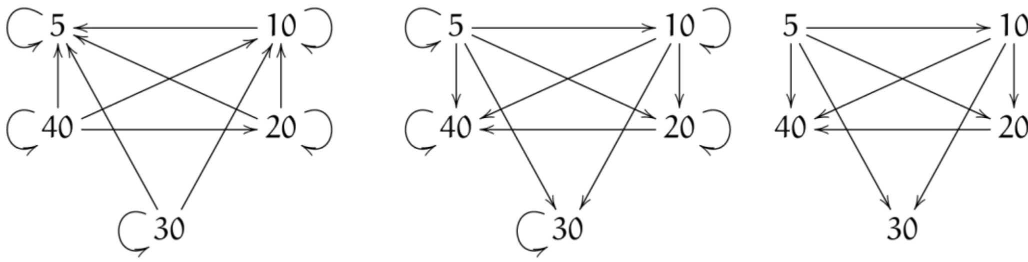


Рисунок 3.8

3. На множині $M = \{1,2,3, 4\}$ задане бінарне відношення:

$$R = \{(1,2),(1,3),(1,1),(2,1),(2,3),(3,4),(4,4)\}.$$

Дослідити відношення R на рефлексивність, симетричність та транзитивність. Якщо воно не є рефлексивним (симетричним, транзитивним), то якими парами його потрібно доповнити, щоб воно стало рефлексивним (симетричним, транзитивним).

4. Представити за допомогою матриці бінарне відношення T на множині $M = \{a,b,c\}$, задане переліком елементів:

а) $T = \{(a,a),(a,b),(b,b),(b,c),(c,a),(d,d)\}$;

б) $T = \{(a,c),(b,a),(b,d),(c,c),(c,d)\}$;

в) $T = \{(a,b),(b,a),(b,d),(c,a),(d,b),(d,c)\}$;

г) $T = \{(a,d),(b,c),(c,b),(c,c),(d,a),(d,d)\}$.

5. Представити за допомогою графа бінарне відношення T на множині $M = \{a,b,c\}$, задане переліком елементів та визначити його властивості.

а) $T = \{(a,a),(b,b),(b,c),(c,a)\}$;

в) $T = \{(a,a),(a,c),(b,a),(b,b)\}$;

б) $T = \{(b,a),(c,b),(c,c),(c,a)\}$;

г) $T = \{(a,a),(a,b),(a,c),(c,c)\}$.

6. На множині $A = \{5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15\}$ задане бінарне відношення $p = \{(x, y) | x \vdots y\}$. Вкажіть всі елементи бінарного відношення, запишіть його матрицю та намалюйте граф.

7. Вкажіть всі елементи бінарного відношення, запишіть його матрицю та намалюйте граф.

а) $p = \{(x, y) : x < y\}$ на множині $X = \{1, 2, 3, 4, 5\}$;

б) $p = \{(x, y) : y = x + 1\}$ на множині $X = \{1, 2, 3, 4, 5\}$.

8. Для кожного з наступних бінарних відношень з'ясуйте які властивості (рефлексивність, симетричність, антисиметричність, транзитивність) воно має, а які ні.

а) $p = \{(x, y) \in R \times R : y^2 = x^2\}$;

б) $p = \{(x, y) \in R \times R : y^2 + x^2 = 1\}$;

в) $p = \{(x, y) \in R \times R : x \cdot y > 1\}$;

г) $p = \{(x, y) \in R \times R : y = |x|\}$;

д) $p = \{(x, y) \in R \times R : y^2 + y = x^2 + x\}$;

е) $p = \{(x, y) \in Z \times Z : x \leq y + 1\}$;

ж) $p = \{(x, y) \in R \times R : y < 2x\}$;

з) $p = \{(x, y) \in R \times R : y^2 = x^2\}$;

і) $p = \{(x, y) \in R \times R : x - y = 12\}$;

к) $p = \{(x, y) \in Z \times Z : (x - y) \vdots 3\}$.

9. Встановити, чи є бінарне відношення T на множині $M = \{1, 2, 3, 4\}$ рефлексивним, антирефлексивним або не рефлексивним.

а) $T = \{(1,1), (2,3), (4,2), (4,1), (4,4)\}$;

б) $T = \{(1,1), (1,2), (2,2), (3,3), (4,4)\}$;

в) $T = \{(1,2), (3,1), (3,2), (3,4), (4,2)\}$;

г) $T = \{(1,1), (1, 4), (2, 2), (3, 2), (3,3), (4, 4)\}$.

10. Встановити, чи є симетричним, антисиметричним або не симетричним бінарне відношення T на множині $M = \{a, b, c, d\}$.

а) $T = \{(a,b), (b,c), (b,d), (c,b), (d,b)\}$;

б) $T=\{(a,b),(b,a),(b,b),(b,c),(d,b)\}$;

в) $T=\{(a,b),(b,a),(b,c),(c,d),(d,c)\}$;

г) $T=\{(b,a),(b,c),(c,a),(c,c),(d,a)\}$.

11. Встановити, чи є транзитивним, або не транзитивним бінарне відношення T на множині $M=\{1,2,3,4\}$.

а) $T=\{(1,1),(1,2),(2,4),(4,3)\}$;

б) $T=\{(1,2),(3,1),(3,2),(4,4)\}$;

в) $T=\{(1,2),(3,1),(3,2),(3,4),(4,2)\}$;

г) $T=\{(1,3),(2,1),(2,3),(3,4)\}$.

12. Задати за допомогою матриці та графа бінарне відношення на множині $M=\{a,b,c,d\}$, яке є одночасно

а) рефлексивним, симетричним та не транзитивним;

б) антирефлексивним, антисиметричним та транзитивним;

в) не рефлексивним, симетричним та не транзитивним;

г) рефлексивним, антисиметричним та не транзитивним;

д) антирефлексивним, не симетричним та не транзитивним;

е) не рефлексивним, не симетричним та транзитивним;

ж) не рефлексивним, симетричним та транзитивним;

з) рефлексивним, не симетричним та не транзитивним.

13. Бінарні відношення $R_1=\{(x,y): x^2+y^2\geq 10\}$ та $R_2=\{(x,y): x^2-y^2\leq 0\}$ задані на множині $U=\{a,b,c,d\}$. Запишіть матриці цих відношень, намалюйте їх графи та з’ясуйте властивості.

14. Знайдіть всі відношення на двохелементній множині $\{a,b,c,d\}$. Вкажіть серед них:

а) всі рефлексивні;

б) всі симетричні;

в) всі антисиметричні;

г) всі транзитивні.

ТЕМА 4.

ВІДНОШЕННЯ ЕКВІВАЛЕНТНОСТІ, ПОРЯДКУ ТА ТОЛЕРАНТНОСТІ.

Ключові слова: відношення еквівалентності, клас еквівалентності, відношення порядку, впорядкована множина, відношення толерантності.

14.1. Основні теоретичні відомості.

Бінарне відношення R на множині M називають *відношенням еквівалентності*, якщо воно рефлексивне, симетричне і транзитивне.

Якщо два елементи множини M знаходяться у відношенні еквівалентності, то їх називають *еквівалентними*. Множину всіх тих елементів множини M, які еквівалентні елементу $a \in M$ називають *класом еквівалентності* цього елемента і позначають $[a]$. Неважко довести, що класи еквівалентності $[a]$, $[b]$ або збігаються, або не перетинаються. Отже, множина M розбивається на класи еквівалентності, які не перетинаються. Навпаки, всяке розбиття множини M на не перетинаючи підмножини визначає між елементами цієї множини деяке відношення еквівалентності.

Відношення строгого порядку ($a < b$) – це бінарне відношення, яке антирефлексивне, антисиметричне і транзитивне.

Відношення нестрогого порядку ($a \leq b$) – це бінарне відношення, що є рефлексивним, антисиметричним і транзитивним.

Якщо для відношення порядку R на множині M і певних відмінних елементів $a, b \in M$ виконується хоча б одне з відношень aRb чи bRa, то елементи a і b називаються *порівнянними*; інакше – *непорівнянними.*

Лінійно впорядкована множина M – це множина, на якій задано відношення порядку, причому будь-які два елементи множини M порівнянні; *частково впорядкована множина* – те ж саме, але допускаються непорівнянні пари елементів.

Відношення толерантності – це бінарне відношення □ на множині M, що є одночасно рефлексивним і симетричним. Це відношення є експлікацією інтуїтивного уявлення про подібність (або схожість) об’єктів. Кожен об'єкт подібний сам з собою (транзитивність), а схожість двох об'єктів не залежить від того, в якому порядку вони порівнюються (симетричність). Якщо один об’єкт схожий с другим, а другий схожий с третім, то це зовсім не означає, що всі вони схожі між собою, так як властивість транзитивності може не виконуватися.

14.2. Приклади виконання завдань.

Приклад 4.1. Наведемо приклади відношення еквівалентності:
1) відношення рівності, паралельності прямих;
2) відношення подібності трикутників;
3) відношення між елементами множини всіх багатокутників: «мати однакове число сторін»;
4) відношення пропорційності P між парами чисел (x, y) та (z, t), якщо $\frac{x}{y}=\frac{z}{t}$;
5) відношення між цілими числами «мати однакові залишки від поділення на m».

Приклад 4.2. Нехай m – деяке натуральне число. Перевіримо, чи є відношенням еквівалентності наступне бінарне відношення на множині цілих чисел $\square = \{(x, y)\in Z \times Z : (x-y)\vdots\, m\}$.

Рефлексивність: для довільного $x \in Z : x - x = 0 = 0\cdot m \Rightarrow (x, x)\in \square$.

Симетричність: нехай $(x,y)\in \square$, тоді $\exists k \in Z$, таке що $x - y = k \cdot m$. Але, $y - x = -k \cdot m \Rightarrow (y, x)\in \square$.

Транзитивність: Нехай $(x,y)\in \square, (y,z)\in \square$, тоді $\exists k, n \in Z$, такі що $x - y = k\cdot m,\ y - z = n\cdot m$. Тоді, $x - z = (k+n)\cdot m \Rightarrow (x, z)\in \square$.

Отже, це відношення є відношенням еквівалентності.

Приклад 4.3. Складемо мінімальне за кількістю елементів відношення еквівалентності на множині $A = \{1, 2, 3, 4, 5\}$ таким чином, щоб $(1,2)\in \square$ та $(2,3)\in \square$.

Відношення еквівалентності рефлексивне, тому відношенню обов`язково повинні належати пари $(1,1),(2,2),(3,3),(4,4),(5,5)$. Відношення еквівалентності симетрично, тому разом з парами $(1,2), (2,3)$ відношенню повинні належати й пари $(2,1), (3,2)$. В силу транзитивності та симетричності повинні бути пари $(3,1)$ та $(1,3)$. Таким чином, мінімальне відношення еквівалентності, яке можна побудувати, має вид:

$$\square = \{(1,1),(2,2),(3,3),(4,4),(5,5),(1,2),(2,1),(2,3),(3,2),(3,1),(1,3)\}.$$

Приклад 4.4. Нехай розбиття $M = \{\{1\},\{2,5\},\{3\},\{4,6,7\}\}$ множини $A = \{1,2,3,4,5,6,7\}$ є класами еквівалентності для деякого відношення еквівалентності . Тоді, = {(1, 1), (2, 2), (5, 5), (2, 5), (5, 2), (3, 3), (4, 4), (6, 6), (7, 7), (4, 6), (6, 4), (4, 7), (7, 4), (6, 7), (7, 6)}.

Приклад 4.5. Відношення $\square = \{(x,y): x \le y\}$ на множині натуральних чисел є відношенням лінійного порядку.

Рефлексивність: $\forall x \in M, \quad x = x \quad \Rightarrow (x, x) \in \square$.

Антисиметричність: нехай $(x, y) \in \square$ та $(y,x) \in \square \Rightarrow \begin{cases} x \le y \\ y \le x \end{cases} \Rightarrow x = y$.

Транзитивність: нехай $(x,y) \in \square$ та $(y,z) \in \square \Rightarrow \begin{cases} x \le y \\ y \le z \end{cases} \Rightarrow x \le z \Rightarrow (x,z) \in \square$.

Крім того, для будь-яких x, y виконується: $x \le y$, або $y \le x$.

14.3. Завдання для самостійного виконання.

1. Покажіть, що кожне з наступних відношень є еквівалентністю
а) подібність на множині всіх трикутників на площині;
б) належність до однієї групи на множині студентів факультету;
в) рівність ваги на множині різноваг;
г) взаємозамінюваність на множині деталей;
д) концентричність на множині кіл на площині;
е) $\{(x,y) \in Z^2 : x \le y\}$.

2. Показати, що відношення $\{(x,y) \in Z^2 : (x-y) \vdots 3\}$ є відношення еквівалентності. Описати класи еквівалентності.

3. На множині R задане бінарне відношення: $x \square y \Leftrightarrow (xy = 1 \vee x = y)$. Чи є це відношення відношенням еквівалентності? Якщо так, то описати його класи еквівалентності.

4. Нехай та $\square$ – відношення еквівалентності на множині X. Довести або спростувати, що $\square \cup \square$, $\square \cap \square$ є відношеннями еквівалентності.

5. Нехай задана множина $A=\{1,2,3,4,5,6\}$ розбиттям на класи: $A_1=\{1,2\}$, $A_2=\{3\}$, $A_3=\{4,5,6\}$. Написати множину всіх пар $A\times A$, які належать відповідному відношенню еквівалентності.

6. Відношення еквівалентності на множині $M=\{1,2,3,4,5,6,7\}$ подано розбиттям на класи: $M_1=\{1,4\}$; $M_2=\{2,3,7\}$; $M_3=\{2,3,7\}$. Представити це відношення множиною впорядкованих пар, матрицею та графом.

7. Встановити, чи є задане за допомогою матриці бінарне відношення на множині $M=\{a,b,c,d\}$ відношенням еквівалентності.

а) $\begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}$ б) $\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$ в) $\begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$ г) $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{pmatrix}$,

д) $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$ е) $\begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}$ ж) $\begin{pmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}$, з) $\begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}$.

8. Визначити число n класів еквівалентності й клас еквівалентності $K(a)$ елемента a, який заданий на множині $M=\{a,b,c,d\}$ бінарним відношенням еквівалентності T.

а) $T=\{(a,a),(b,b),(c,c),(c,d),(d,c),(d,d)\}$;

б) $T=\{(a,a),(a,b),(a,c),(b,b),(b,a),(b,c),(c,c),(c,a),(c,b),(d,d)\}$;

в) $T=\{(a,a),(a,d),(b,b),(b,c),(c,b),(c,c),(d,a),(d,d)\}$;

г) $T=\{(a,a),(b,b),(c,c),(d,d)\}$;

д) $T=\{(a,a),(b,b),(b,c),(c,c),(c,b),(d,d)\}$;

е) $T=\{(a,a),(a,b),(a,c),(a,d),(b,a),(b,b),(b,c),(b,d),(c,c),(c,a),(c,b),(c,d),$ $(d,a),(d,b),(d,c),(d,d)\}$;

ж) $T=\{(a,a),(a,c),(a,d),(b,d),(c,c),(c,a),(c,d),(d,a),(d,c),(d,d)\}$;

з) $T=\{(a,a),(a,c),(b,b),(c,c),(c,a),(d,d)\}$.

9. Показати, що наведені нижче відношення є відношеннями порядку й визначити тип впорядкованості.

а) «x важча за y» на множині деталей;

б) «x підпорядкований y» на множині посад;

в) «x довший y» на множині відрізків на площині;

г) «x старший y» на множині людей;

д) «x не перевершує y» на множині номерів будинків на вулиці;

е) «з x витікає y» на множині висловів;

ж) «x знаходиться всередині y» на множині кіл на площині;

з) « $x \le y$ » на множині натуральних чисел Z;

і) « $x^2 \le y^2$» на множині Z;

к) «x ділиться на y» на множині Z.

10. Визначити, чи є задане на множині $M = \{1, 2, 3, 4\}$ бінарне відношення T бінарним відношення впорядкованості « $\le$».

а) $T = \{(1,1),(1,2),(2,2),(3,1),(3,2),(3,3),(4,4)\}$;

б) $T = \{(1,1),(1,4),(2,2),(3,1),(3,4),(4,2),(4,4)\}$;

в) $T = \{(1,1),(2,2),(2,3),(2,4),(3,3),(3,4),(4,3),(4,4)\}$;

г) $T = \{(1,1),(1,3),(2,1),(2,2),(2,3),(2,4),(3,3),(4,3),(4,4)\}$;

д) $T = \{(1,1),(2,2),(2,4),(3,2),(3,3),(4,1),(4,4)\}$;

е) $T = \{(1,1),(1,3),(2,2),(2,4),(3,1),(3,3),(4,2),(4,4)\}$;

ж) $T = \{(1,1),(1,2),(2,2),(3,1),(3,2),(3,3),(3,4),(4,4)\}$;

з) $T = \{(1,1),(1,2),(1,4),(2,2),(3,1),(3,2),(3,3),(3,4),(4,2),(4,4)\}$.

11. Встановити, чи є задане за допомогою матриці бінарне відношення на множині $M = \{a, b, c, d\}$ відношенням строгої впорядкованості.

а) $\begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$ б) $\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix}$ в) $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{pmatrix}$ г) $\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix}$,

д) $\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}$ е) $\begin{pmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{pmatrix}$ ж) $\begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$ з) $\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$.

12. Побудуйте граф відношення ділимості на множині {2, 3, 4, 6, 8, 9, 12, 18, 24, 36}.

13. На множині натуральних чисел Z задано відношення:

$$(x_1, y_1)R(x_2, y_2) \Leftrightarrow x_1 \le x_2,\ y_1 \le y_2 .$$

Доведіть, що це відношення порядку. Знайдіть всі мінімальні та максимальні відносно R елементи в множинах:

а) $\{(x,y): x \le 3, y \le 4\}$;

б) $\{(x,y): 2 \le x + y \le 4\}$;

в) $\{(x,y): x^2 + y^2 \le 4\}$.

14. Знайдіть всі відношення на двохелементній множині $\{a,b\}$. Вкажіть серед них:

а) всі відношення еквівалентності;

б) всі відношення порядку.

15. Яке з приведених нижче відношень є толерантними?

а) «x перпендикулярна y» на множині прямих;

б) «x знайомий з y» на множині людей;

в) «x має спільні точки з y» на множині геометричних фігур;

г) «x поряд з y» на множині книг на полці.

16. На множині прямих $U = \{a, b, c, d, l\}$ задано відношення «пряма p паралельна або співпадає з прямою q», $p, q \in U$. Перевірити, чи є задане відношення відношенням еквівалентності, порядку або толерантності.

$$a: 2x - 3y + 4 = 0,\ b: 2x + 3y + 1 = 0,\ c: -4x + 6y + 5 = 0,$$
$$d: 4x - 3y - 5 = 0,\ l: 3 - 10x - 15y = 0.$$

ТЕМА 5.

ТАБЛИЦІ ІСТИННОСТІ БУЛЕВИХ ФУНКЦІЙ.

Ключові слова: булева (логічна) функція, логічний вираз, таблиця істинності.

5.1. Основні теоретичні відомості.

Булева (логічна) функція – це функція, область значень якої 0 та 1, та яка залежить від змінних, що набувають лише цих значень.

Таблиця 5.1 – Елементарні булеві функції двох змінних

Позначення функції	Назва функції	x_1			
		0	0	1	1
		x_2			
		0	1	0	1
$y_0 = 0$	константа 0	0	0	0	0
$y_1 = x_1 \wedge x_2$	кон’юнкція	0	0	0	1
$y_2 = x_1 - x_2$	коімплікація ліва	0	0	1	0
$y_3 = x_1$		0	0	1	1
$y_4 = x_2 - x_1$	коімплікація права	0	1	0	0
$y_5 = x_2$		0	1	0	1
$y_6 = x_1 + x_2$	сума по модулю 2	0	1	1	0
$y_7 = x_1 \vee x_2$	диз’юнкція	0	1	1	1
$y_8 = x_1 \downarrow x_2$	стрілка Пірса	1	0	0	0
$y_9 = x_1 \leftrightarrow x_2$	еквівалентність	1	0	0	1
$y_{10} = \bar{x}_2 = \rceil x_2$	заперечення	1	0	1	0
$y_{11} = x_1 \leftarrow x_2$	імплікація права	1	0	1	1
$y_{12} = \bar{x}_1 = \rceil x_1$	заперечення	1	1	0	0
$y_{13} = x_1 \rightarrow x_2$	імплікація ліва	1	1	0	1
$y_{14} = x_1 \mid x_2$	штрих Шеффера	1	1	1	0
$y_{15} = 1$	константа 1	1	1	1	1

Відомо, що кількість всіх можливих елементарних булевих функцій від n змінних дорівнює 2^{2^n}. Отже, існує лише 16 елементарних булевих функцій від двох змінних, які визначають основні логічні операції.

Між функціями таблиці є така залежність:

$$y_i = \overline{y_{15-i}} \quad (i = 0, 1, ..., 15).$$

З 16 функцій 6 тривіальних – y_0, y_3, y_5, y_{10}, y_{12}, y_{15}. З тих десяти, що залишилися, – y_4 та y_{11} не є самостійними, тому що відрізняються від відповідних їм y_2 і y_{13} лише порядком розташування аргументів. Ті вісім, що залишилися, – y_1, y_2, y_6, y_7, y_8, y_9, y_{13}, y_{14} є оригінальними.

Між оригінальними булевими функціями є теж взаємозв’язок. Перш за все, мають місце закони Де Моргана, які пов’язують кон’юнкцію та диз’юнкцію:

$$\overline{x_1 \vee x_2} = \overline{x_1} \wedge \overline{x_2};$$
$$\overline{x_1 \wedge x_2} = \overline{x_1} \vee \overline{x_2}.$$

Якщо обмежитись таким набором функцій, як *константа, заперечення, кон'юнкція* та *диз'юнкція*, то будь-яка інша функція може бути записана через ці функції. З урахуванням законів Де Моргана випливає, що будь-яка булева функція може бути виражена через заперечення й кон'юнкцію або через заперечення й диз'юнкцію.

Вкажемо деякі відомі тотожності:

$$x_1 - x_2 = x_1 \wedge \overline{x_2};$$
$$x_2 - x_1 = \overline{x_1} \wedge x_2;$$
$$x_1 + x_2 = \overline{x_1} \wedge x_2 \vee x_1 \wedge \overline{x_2};$$
$$x_1 \downarrow x_2 = \overline{x_1 \vee x_2};$$
$$x_1 \leftrightarrow x_2 = \overline{x_1} \wedge \overline{x_2} \vee x_1 \wedge x_2;$$
$$x_1 \leftarrow x_2 = \bar{x}_2 \vee x_1;$$
$$x_1 \rightarrow x_2 = \bar{x}_1 \vee x_2;$$
$$x_1 \mid x_2 = \overline{x_1} \vee \overline{x_2}.$$

Досить корисним для розуміння взаємозалежності булевих функцій є наступне:

$$x_1 \wedge x_2 = 1 \Leftrightarrow x_1 = x_2 = 1;$$
$$x_1 - x_2 = 1 \Leftrightarrow x_1 = 1, x_2 = 0;$$
$$x_2 - x_1 = 1 \Leftrightarrow x_1 = 0, x_2 = 1;$$
$$x_1 + x_2 = 1 \Leftrightarrow x_1 \neq x_2;$$
$$x_1 \vee x_2 = 0 \Leftrightarrow x_1 = x_2 = 0;$$
$$x_1 \downarrow x_2 = 1 \Leftrightarrow x_1 = x_1 = 0;$$
$$x_1 \leftrightarrow x_2 = 1 \Leftrightarrow x_1 = x_2;$$
$$x_1 \leftarrow x_2 = 0 \Leftrightarrow x_1 = 0, x_2 = 1;$$
$$x_1 \rightarrow x_2 = 0 \Leftrightarrow x_1 = 1, x_2 = 0;$$
$$x_1 | x_2 = 0 \Leftrightarrow x_1 = x_2 = 1.$$

Приведені властивості функцій вказують на те, що коімплікація є запереченням імплікації, сума по модулю 2 є запереченням еквівалентності, штрих Шеффера є запереченням кон’юнкції, стрілка Пірса є запереченням диз’юнкції. Цей факт можна прослідити на значеннях функції безпосередньо в табл. 5.1. Наприклад,

x_1	0	0	1	1
x_2	0	1	0	1
$x_1 \vee x_2$	0	1	1	1
$x_1 \downarrow x_2$	1	0	0	0

Будь-яка композиція логічних змінних за допомогою розглянутих логічних операцій вважається складною булевою функцією, або логічним виразом. Щоб зменшити кількість дужок у формулах складних булевих функцій вводять пріоритет операцій: 1) заперечення; 2) кон’юнкція; 3) усі інші операції.

5.2. Приклади виконання завдань.

Приклад 5.1. Побудувати таблицю істинності для функції

$$(x_1 \vee x_2) \wedge (x_2 \rightarrow \bar{x}_3).$$

x_1	0	0	0	0	1	1	1	1
x_2	0	0	1	1	0	0	1	1
x_3	0	1	0	1	0	1	0	1
$x_1 \vee x_2$	0	0	1	1	1	1	1	1
$\bar{x}_3$	1	0	1	0	1	0	1	0
$x_2 \to \bar{x}_3$	1	1	1	0	1	1	1	0
$(x_1 \vee x_2) \wedge (x_2 \to \bar{x}_3)$	0	0	1	0	1	1	1	0

Приклад 5.2. Побудувати таблицю істинності для функції

$$\overline{x_1 \to \overline{(x_2 \wedge x_1)}} \to x_1 \vee x_3$$

x_1	0	0	0	0	1	1	1	1
x_2	0	0	1	1	0	0	1	1
x_3	0	1	0	1	0	1	0	1
$x_2 \wedge x_1$	0	0	0	0	0	0	1	1
$\overline{x_2 \wedge x_1}$	1	1	1	1	1	1	0	0
$x_1 \to (\overline{x_2 \wedge x_1})$	1	1	1	1	1	1	0	0
$\overline{x_1 \to \overline{(x_2 \wedge x_1)}}$	0	0	0	0	0	0	1	1
$\overline{x_1 \to \overline{(x_2 \wedge x_1)}} \to x_1$	1	1	1	1	1	1	1	1
$\overline{x_1 \to \overline{(x_2 \wedge x_1)}} \to x_1 \vee x_3$	1	1	1	1	1	1	1	1

Отже, ця булева функція представляє собою тотожну одиницю.

Приклад 5.3. Побудувати таблицю істинності для логічного виразу

$$A \vee \neg(B \wedge \overline{C}) \vee \neg(A \vee B \vee C)$$

A	0	0	0	0	1	1	1	1
B	0	0	1	1	0	0	1	1
C	0	1	0	1	0	1	0	1
$\overline{C}$	1	0	1	0	1	0	1	0
$B \wedge \overline{C}$	0	0	1	0	0	0	1	0
$\neg(B \wedge \overline{C})$	1	1	0	1	1	1	0	1

A	0	0	0	0	1	1	1	1
B	0	0	1	1	0	0	1	1
C	0	1	0	1	0	1	0	1
$A \vee \neg(B \wedge \overline{C})$	1	1	0	1	1	1	1	1
$\overline{A}$	1	1	1	1	0	0	0	0
$\overline{A} \vee B \vee \overline{C}$	1	1	1	1	1	0	1	1
$\neg(\overline{A} \vee B \vee \overline{C})$	0	0	0	0	0	1	0	0
$A \vee \neg(B \wedge \overline{C}) \vee \neg(\overline{A} \vee B \vee \overline{C})$	1	1	0	1	1	1	1	1

5.3. Завдання для самостійного виконання.

Побудувати таблиці істинності для логічних виразів:

1. $(A \to B) \leftrightarrow (\overline{A \vee B})$.
2. $(A \to B \vee C) \wedge \overline{A \wedge C} \to A$.
3. $\overline{A} \leftrightarrow B$.
4. $(A \to B) \wedge (C \to D)$.
5. $A \to \overline{B}$.
6. $(A \to B) \leftrightarrow (\overline{B} \wedge C) \vee A$.
7. $\overline{A} \vee B$.
8. $A \to (B \wedge C) \leftrightarrow (A \vee \overline{B}) \wedge (A \vee \overline{C})$.
9. $\overline{B \leftrightarrow A}$.
10. $\neg(A \to \overline{(B \wedge A)}) \to A \vee C$.
11. $\overline{\overline{A} \vee B}$.
12. $A \wedge (B \to A) \to \overline{A}$.
13. $\overline{A \leftrightarrow \overline{B}}$.
14. $(A \wedge \overline{B} \to B) \to (A \to B)$.
15. $\overline{A \to \overline{B}}$.
16. $A \wedge (B \vee \overline{A}) \wedge (\overline{B} \to A) \vee B$.
17. $(A \wedge \overline{B}) \to B$.
18. $(A \to B) \wedge \overline{A} \to \overline{B}$.
19. $(A \vee \overline{B}) \to \overline{A}$.
20. $\overline{A} \wedge B \to A \vee B$.
21. $\overline{A} \to A \vee B$.
22. $A \to B \leftrightarrow \overline{A} \vee B$.
23. $(A \vee B) \wedge (\overline{A} \vee \overline{B})$.
24. $A \to \neg(B \vee C)$.
25. $((A \wedge B) \to C) \leftrightarrow (A \to (\overline{B} \vee C))$.
26. $A \to (A \to B)$.
27. $A \to (B \vee C) \leftrightarrow (A \to B) \vee (A \to C)$.
28. $(A \vee B \to \overline{C}) \to A$.
29. $(A \wedge B) \leftrightarrow (B \wedge \overline{C})$.
30. $\overline{B} \to \overline{A}$.

ТЕМА 6.

АЛГЕБРА БУЛЯ.

Ключові слова: алгебра Буля, принцип подвійності, двоїста формула.

6.1. Основні теоретичні відомості.

Множина всіх булевих функцій з операціями заперечення, кон'юнкції й диз'юнкції утворює *алгебру Буля*.

Властивості булевої алгебри:

комутативний закон:

1а) $x_1 \vee x_2 = x_2 \vee x_1$ 1б) $x_1 \wedge x_2 = x_2 \wedge x_1$

асоціативний закон:

2а) $x_1 \vee (x_2 \vee x_3) = (x_1 \vee x_2) \vee x_3$ 2б) $x_1 \wedge (x_2 \wedge x_3) = (x_1 \wedge x_2) \wedge x_3$

дистрибутивний закон:

3а) $x_1 \vee (x_2 \wedge x_3) = (x_1 \vee x_2) \wedge (x_1 \vee x_3)$ 3б) $x_1 \wedge (x_2 \vee x_3) = (x_1 \wedge x_2) \vee (x_1 \wedge x_3)$

властивості констант:

4а) $x \vee 0 = x$ 4б) $x \wedge 1 = x$

властивість заперечення:

5а) $x \vee \bar{x} = 1$ 5б) $x \wedge \bar{x} = 0$

6а) $x \vee 1 = 1$ 6б) $x \wedge 0 = 0$

7а) $\bar{0} = 1$ 7б) $\bar{1} = 0$

закон ідемпотентності (самопоглинання):

8а) $x \vee x = x$ 8б) $x \wedge x = x$

закон поглинання:

9а) $x_1 \vee (x_1 \wedge x_2) = x_1$ 9б) $x_1 \wedge (x_1 \vee x_2) = x_1$

закон склеювання:

10а) $(a \vee b) \wedge (a \vee \bar{b}) = a$ 10б) $(a \wedge b) \vee (a \wedge \bar{b}) = a$

закон Де-Моргана:

11а) $\overline{x_1 \vee x_2} = \bar{x}_1 \wedge \bar{x}_2$ 11б) $\overline{x_1 \wedge x_2} = \bar{x}_1 \vee \bar{x}_2$

В булевій алгебрі, як і в алгебрі множин, має місце *принцип подвійності.* Взаємно двоїстими операціями є диз'юнкція й кон'юнкція. Заміняючи в деякому логічному виразі кожну операцію на двоїсту їй, одержуємо *двоїстий вираз.*

6.2. Приклади виконання завдань.

Приклад 6.1. Довести тотожність:

$$((a+b)\to(a\vee b))\wedge((a\vee b)\to(a+b)) = a\mid b.$$

Доведення тотожності проводимо за допомогою дистрибутивного закону (3а), закону властивості константи (4б), закону поглинання та наступних співвідношень функцій: $a\to b=\bar a\vee b,\ \ a+b=(a\vee b)\wedge(\bar a\vee\bar b)$.

$$((a+b)\to(a\vee b))\wedge((a\vee b)\to(a+b))=(((\bar a\vee\bar b)\wedge(a\vee b))\to(a\vee b))\wedge$$
$$\wedge((a\vee b)\to((\bar a\vee\bar b)\wedge(a\vee b)))=$$

/використовуємо $a\to b=\bar a\vee b$/

$$=((a\wedge b)\vee\overline{(a\vee b)}\vee a\vee b)\wedge(\overline{(a\vee b)}\vee(\bar a\wedge(a\vee b))\wedge(\bar b\wedge(a\vee b)))=$$

за допомогою закону поглинання та дистрибутивному закону:

$$=((a\wedge\bar b)\vee\bar a\vee b)\wedge((\bar a\wedge\bar b)\vee(a\wedge\bar a)\vee(a\wedge\bar b)\vee(b\wedge\bar a)\vee(\bar b\wedge b))=$$
$$=(((\bar a\vee a)\wedge(\bar b\vee a))\vee b)\wedge((\bar a\wedge\bar b)\vee(a\wedge\bar b)\vee(\bar a\wedge b))=$$
$$=(\bar b\vee a\vee b)\wedge(\bar b\vee(b\wedge\bar a))=1\wedge(\bar b\vee\bar a)=\bar a\vee\bar b=a|b.$$

Приклад 6.2. Довести тотожність: $a\to(b\wedge c)=a|(b|c)$.

Використовуємо функції $x_1\to x_2=\bar x_1\vee x_2$, $x_1\wedge x_2=\overline{\bar x_1\vee\bar x_2}$, $\bar x_1\vee\bar x_2=x_1|x_2$.

$$a\to(b\wedge c)=\bar a\vee\left(\overline{\overline{b\wedge c}}\right)=\bar a\vee\left(\overline{b|c}\right)=a|(b|c).$$

Приклад 6.3. Довести тотожність:

$$(a\downarrow b)\to(a+b))\wedge((a-b)\nrightarrow(ab))=a\vee b.$$

$$(a\downarrow b)\to(a+b))\wedge((a-b)\nrightarrow(ab))=((a\vee b)\vee(\overline{(a\wedge b)}\vee(a\vee b))\wedge$$
$$\wedge((\bar a\vee b)\vee(\bar a\vee\bar b))=((a\vee b)\wedge((a\vee b)\vee(\bar a\vee\bar b)))\wedge(a\vee 1)=(x\wedge(x\vee y))\wedge 1=x,$$

де $x=a\vee b, y=\bar a\vee\bar b$.

6.3. Завдання для самостійного виконання.

Довести тотожність або спростити логічний вираз до операцій булевої алгебри, використовуючи основні властивості операцій та закони алгебри Буля. Результат перевірити за допомогою таблиць істинності.

1. $(A\to B)\to(\overline{B}\leftrightarrow\overline{A})$.
2. $a+(b\wedge c)=(a\to b)\leftrightarrow((a+c)\wedge b)$.
3. $((a\downarrow b)\to(a+b))\wedge((a-b)\to(a\mid b))=a\vee b$.
4. $(a\vee\bar{c})\wedge(\bar{a}\vee\bar{b})\wedge(\bar{b}\vee c)\wedge(\bar{a}\vee b)\wedge(b\vee c)=0$.
5. $(a\vee b\vee c)+(a\wedge b\wedge c)=(a\vee b\vee c)-(a\wedge b\wedge c)$.
6. $((\bar{a}\vee b)\wedge(b\vee c))\vee(a\vee\bar{c})\wedge(\bar{b}\vee c)\wedge(\bar{a}\vee\bar{b})=1$.
7. $(a\wedge b\wedge c)\to(a\vee b\vee c)=(a\to b)\vee(b\to c)\vee(c\to a)$.
8. $(a/b)/c-c=a\mid(b\mid c-a)$.
9. $x\wedge z\vee y\wedge\bar{z}\vee x\wedge y=x\wedge z\vee y\wedge\bar{z}$.
10. $x\vee\bar{x}\wedge y=x\wedge y\vee x\wedge\bar{y}\vee\bar{x}\wedge y=x\vee y$.
11. $x\wedge y\vee x\wedge\bar{y}=x$.
12. $(a\downarrow b)\downarrow(a\downarrow b)=a\vee b$.
13. $(a\downarrow a)\downarrow(b\downarrow b)=a\wedge b$.
14. $\overline{a\leftrightarrow b}\vee a+b$.
15. $a\leftrightarrow(b\mid c)=(a\to b)\leftrightarrow(a+c)\wedge b$.
16. $((a\mid b)\to(b\vee c))\vee(c\leftrightarrow d)=d\to(c\vee b)$.
17. $(a\downarrow b)\downarrow(\bar{a}\mid c)\downarrow(\bar{b}\mid d)=a\vee b$.
18. $a+(c-b)=(a\leftrightarrow c)+(b\mid c)$.
19. $(a-b)+(a+c)\wedge b=\bar{a}+(b\mid c)$.
20. $a+b+c+d=((a\leftrightarrow b)\to(c+d))\wedge((a+b)\mid(c+d))$.
21. $(\overline{(p\vee q)\wedge r)}\wedge\overline{(s\vee r)\wedge(\overline{p\vee q\vee r)}}\wedge t\wedge\bar{t}$.
22. $p\wedge q\wedge(s\to(s\vee t))$.
23. $(A_1\wedge A_3)\vee(A_1\to\bar{A}_3)\vee(A_2\wedge A_3)\overline{\vee}(A_1\wedge A_2\wedge A_3)$.
24. $(A_1\to A_2)\wedge(A_2\to A_3)\to(A_3\to A_1)$.
25. $p\wedge(q\vee\bar{p})\wedge((\bar{q}\to p)\vee q)$.
26. $(A\wedge(A\vee B))\wedge(\overline{B}\to A)$.
27. $((A\to B)\wedge C)\vee\overline{A}\wedge B$.

28. $A \wedge B \to (\overline{B} \wedge B \to C)$.

29. $(A \leftrightarrow B) \wedge (A\overline{B} \vee \overline{A}B)$.

30. $\overline{\overline{A}\,\overline{B}} \vee (A \to B) \wedge A$.

ТЕМА 7.

ДИЗ'ЮНКТИВНА НОРМАЛЬНА ФОРМА.

Ключові слова: конституента, диз'юнктивна нормальна форма, кон'юнктивна нормальна форма, досконала нормальна форма, конституента одиниці, конституента нуля, терм.

7.1. Основні теоретичні відомості.

Будь-яку булеву функцію $y = f(a,b)$ можна представити як деяку комбінацію областей: $C_0 = \overline{a} \wedge \overline{b}$, $C_1 = a \wedge \overline{b}$, $C_2 = \overline{a} \wedge b$, $C_3 = a \wedge b$.

Тоді, залежно від значення функції й заданих C_i , які в цьому випадку ми будемо йменувати *конституентами*, одержимо шістнадцять логічних операцій:

$$y = \left[\overline{a} \wedge \overline{b} \wedge f(0,0)\right] \vee \left[a \wedge \overline{b} \wedge f(1,0)\right] \vee \left[\overline{a} \wedge b \wedge f(0,1)\right] \vee \left[a \wedge b \wedge f(1,1)\right].$$

Подібна форма представлення логічних функцій називається *досконалою диз'юнктивною нормальною формою* (ДДНФ).

У логіці Буля діє *принцип подвійності*: при одночасній заміні символів $\wedge \Leftrightarrow \vee$ та $1 \Leftrightarrow 0$ усі логічні рівності залишаються в силі. Тому ДДНФ може бути перетворена у вираз:

$$y = \left[\overline{a} \vee \overline{b} \vee f(1,1)\right] \wedge \left[a \vee \overline{b} \vee f(0,1)\right] \wedge \left[\overline{a} \vee b \vee f(1,0)\right] \wedge \left[a \vee b \vee f(0,0)\right],$$

який називається *досконалою кон'юнктивною нормальною формою* (ДКНФ). Тут уже конституенти представлені не у вигляді *кон'юнктів*, як у ДДНФ, а у вигляді *диз'юнктів*.

Існує і третя форма – *досконала поліноміальна нормальна форма* (ДПНФ). Її легко одержати зі ДДНФ шляхом заміни $a \vee b = a + b + ab$, $\overline{a} = 1 + a$.

У табл. 7.1 наведено список елементарних логічних функцій від двох аргументів у трьох *досконалих формах* – ДДНФ, ДКНФ і ДПНФ.

Таблиця 7.1

$y = f(a,b)$	**ДДНФ = ДКНФ = ДПНФ**
$y_0 = 0$	$= (a \vee b) \wedge (\bar{a} \vee b) \wedge (a \vee \bar{b}) \wedge (\bar{a} \vee \bar{b}) = 0$
$y_1 = a \wedge b$	$= (a \vee b) \wedge (\bar{a} \vee b) \wedge (a \vee \bar{b}) = ab$
$y_2 = b - a$	$= \bar{a} \wedge b = (a \vee b) \wedge (\bar{a} \vee b) \wedge (\bar{a} \vee \bar{b}) = b + ab$
$y_3 = b$	$= (\bar{a} \wedge b) \vee (a \wedge b) = (a \vee b) \wedge (\bar{a} \vee b) = b$
$y_4 = a - b$	$= a \wedge \bar{b} = (a \vee b) \wedge (a \vee \bar{b}) \wedge (\bar{a} \vee \bar{b}) = a + ab$
$y_5 = a$	$= (a \wedge \bar{b}) \vee (a \wedge b) = (a \vee b) \wedge (a \vee \bar{b}) = a$
$y_6 = a + b$	$= (a \wedge \bar{b}) \vee (\bar{a} \wedge b) = (a \vee b) \wedge (\bar{a} \vee \bar{b}) = a + b$
$y_7 = a \vee b$	$= (a \wedge \bar{b}) \vee (\bar{a} \wedge b) \vee (a \wedge b) = (a \vee b) = a + b + ab$
$y_8 = a \downarrow b$	$= \bar{a} \wedge \bar{b} = (a \vee b) \wedge (\bar{a} \vee b) \wedge (\bar{a} \vee \bar{b}) = 1 + a + b + ab$
$y_9 = a \leftrightarrow b$	$= (\bar{a} \wedge \bar{b}) \vee (a \wedge b) = (\bar{a} \vee b) \wedge (a \vee \bar{b}) = 1 + a + b$
$y_{10} = \bar{a}$	$= (\bar{a} \wedge \bar{b}) \vee (\bar{a} \wedge b) = (\bar{a} \vee b) \wedge (\bar{a} \vee \bar{b}) = 1 + a$
$y_{11} = a \rightarrow b$	$= (\bar{a} \wedge \bar{b}) \vee (\bar{a} \wedge b) \vee (a \vee b) = 1 + a + ab$
$y_{12} = \bar{b}$	$= (\bar{a} \wedge \bar{b}) \vee (a \wedge \bar{b}) = (a \vee \bar{b}) \wedge (\bar{a} \vee \bar{b}) = 1 + b$
$y_{13} = b \rightarrow a$	$= (\bar{a} \wedge \bar{b}) \vee (a \wedge \bar{b}) \vee (a \wedge b) = a \vee \bar{b} = 1 + b + ab$
$y_{14} = a \mid b$	$= (\bar{a} \wedge \bar{b}) \vee (a \wedge \bar{b}) \vee (\bar{a} \wedge b) = 1 + ab$
$y_{15} = 1$	$= (\bar{a} \wedge \bar{b}) \vee (a \wedge \bar{b}) \vee (\bar{a} \wedge b) \vee (a \wedge b) = 1$

Усякий логічний вираз приводиться до нормальної форми наступним шляхом:

1) перетворити вираз таким чином, щоб мати лише операції диз'юнкції, кон'юнкції та заперечення.

2) за допомогою законів Де Моргана логічний вираз перетворити до такого виду, коли знаки заперечення будуть відноситись тільки до окремих змінних;

3) на основі дистрибутивного закону звести до диз'юнкції кон'юнкцій або кон'юнкції диз'юнкцій;

4) отриманий вираз спрощується відповідно до тотожностей ідемпотентності й заперечення.

Якщо в кожному члені нормальної форми представлені всі змінні (або в прямому, або в інверсному виді), то вона називається *досконалою нормальною формою.*

Відомо, що будь-яка булева функція, що не є тотожним нулем (одиницею), має *одну й тільки одну* досконалу диз'юнктивну (кон'юнктивну) нормальну форму.

Для сукупності змінних $x_1, x_2, ..., x_n$ вираз $\tilde{x}_1\tilde{x}_2...\tilde{x}_n$ називають *конституентою одиниці,* а вираз $\tilde{x}_1 \vee \tilde{x}_2 \vee ... \vee \tilde{x}_n$ – *конституентою нуля* , де $\tilde{x}_i$ означає або x_i, або $\bar{x}i$, які називають *термами*. Певна конституента одиниці (нуля) приймає значення одиниця (нуль) тільки при одному відповідному їй наборі значень змінних, який виходить, якщо всі змінні прийняти рівними одиниці (нулю), а їх заперечення – нулю (одиниці). Наприклад, конституенті одиниці $x_1\bar{x}_2x_3x_4$ відповідає набір (1011), а конституенті нуля $\bar{x}_1 \vee x_2 \vee x_3 \vee \bar{x}_4$ – набір (1001).

Так як досконала диз'юнктивна (кон'юнктивна) нормальна форма є диз'юнкцією (кон'юнкцією) конституент одиниці (нуля), то можна затверджувати, що, булева функція $f\ (x_1, x_2, ..., x_n\ )$, що представляється нею, дорівнює одиниці (нулю) тільки при наборах значень змінних $x_1, x_2, ..., x_n$, що відповідають цим конституентам. На інших наборах ця функція обертається в нуль (одиницю).

Навпаки, досконалі форми дозволяють виразити аналітичною формулою будь-яку функцію, якщо відома її таблиця істинності. Для цього необхідно записати диз'юнкції (кон'юнкції) конституент одиниці (нуля), відповідних до наборів значень змінних, на яких функція ухвалює значення, рівне одиниці (нулю).

Наприклад, нехай булеву функцію $u(x, y, z)$ задано таблицею:

x	0	0	0	0	1	1	1	1
y	0	0	1	1	0	0	1	1
z	0	1	0	1	0	1	0	1
u	1	0	1	1	0	1	0	1

Тоді, $u = \bar{x}\bar{y}\bar{z} \vee \bar{x}y\bar{z} \vee \bar{x}yz \vee x\bar{y}z \vee xyz = (\bar{x} \vee \bar{y} \vee z)(x \vee \bar{y} \vee \bar{z})(x \vee y \vee \bar{z})$.

7.2. Приклади виконання завдань.

Приклад 7.1. Вираз $(xy \vee \bar{y}z)\overline{\bar{x}u}$ представити в диз'юнктивній та кон’юнктивній нормальній формі.

$$(xy \vee \bar{y}z)\overline{\bar{x}u} = (xy \vee \bar{y}z)(x \vee \bar{u}) = (xy \vee \bar{y}z)x \vee (xy \vee \bar{y}z)\bar{u} =$$
$$= xyx \vee \bar{y}zx \vee xy\bar{u} \vee y\bar{z}\bar{u} = xy \vee x\bar{y}z \vee xy\bar{u} \vee y\bar{z}\bar{u}$$

(диз'юнктивна нормальна форма) ;

$$(xy \vee y\bar{z})\overline{x\bar{y}} = (xy \vee y\bar{z})(x \vee \bar{u}) = (x \vee y\bar{z})(y \vee \bar{y}z)(x \vee \bar{u}) =$$
$$= (x \vee y)(\bar{x} \vee z)(y \vee y)(y \vee z)(x \vee \bar{u}) = (x \vee \bar{y})(x \vee z)(y \vee z)(\bar{x} \vee u)$$

(кон'юнктивна нормальна форма).

Приклад 7.2. Привести дану функцію до досконалої диз'юнктивної нормальної форми: $\overline{x \to (\bar{x} \sim z)}(y \to \bar{z}) \vee \overline{x \to z}$.

$$\overline{x \to (\bar{x} \sim z)}(y \to \bar{z}) \vee \overline{x \to z} = \bar{\bar{x}} \vee \overline{(x \vee z)(\bar{x} \vee \bar{z})}(\bar{y} \vee \bar{z}) \vee \overline{\bar{x} \vee z} =$$
$$= x\overline{(x \vee z)(\bar{x} \vee \bar{z})}(\bar{y} \vee \bar{z}) \vee x\bar{z} = x(\bar{x}\bar{z} \vee xz)(\bar{y} \vee \bar{z}) \vee x\bar{z} =$$
$$(x\bar{x}\bar{z} \vee xxz)(\bar{y} \vee \bar{z}) \vee x\bar{z} = xz(\bar{y} \vee \bar{z}) \vee \bar{x}z = x\bar{z}y \vee x\bar{z}z \vee \bar{x}z = \bar{x}yz \vee \bar{x}z$$

Третій член диз'юнктивної нормальної форми не містить змінної y, то вона вводиться тотожним перетворенням $(y \vee \bar{y})$. Відповідно

$$x\bar{y}z \vee \bar{x}z = \bar{x}yz \vee \bar{x}z(y \vee \bar{y}) = \bar{x}yz \vee \bar{x}yz \vee \bar{x}\bar{y}z .$$

Приклад 7.3. Привести до досконалої кон'юнктивної нормальної форми.

$$\overline{x \vee y\bar{z}}(x \vee z) = \bar{x}\,\overline{y\bar{z}}(x \vee z) = \bar{x}(\bar{y} \vee z)(x \vee z) = (\bar{x} \vee y\bar{y})(\bar{y} \vee z \vee x\bar{x})(x \vee$$
$$\vee z \vee y\bar{y}) = (\bar{x} \vee y)(\bar{x} \vee \bar{y})(\bar{y} \vee z \vee x)(\bar{y} \vee z \vee \bar{x})(x \vee z \vee y)(x \vee z \vee \bar{y}) =$$
$$= (\bar{x} \vee y \vee z\bar{z}) \wedge (\bar{x} \vee \bar{y} \vee z\bar{z})(x \vee \bar{y} \vee z)(\bar{x} \vee \bar{y} \vee z)(x \vee y \vee z)(x \vee \bar{y} \vee z) =$$
$$= (\bar{x} \vee y \vee z)(\bar{x} \vee y \vee \bar{z})(\bar{x} \vee \bar{y} \vee z) \wedge$$
$$\wedge (\bar{x} \vee \bar{y} \vee z)(x \vee \bar{y} \vee z)(\bar{x} \vee \bar{y} \vee z)(x \vee y \vee z)(x \vee \bar{y} \vee z) =$$
$$= (\bar{x} \vee y \vee z)(\bar{x} \vee y \vee \bar{z}) \wedge (\bar{x} \vee \bar{y} \vee \bar{z})(\bar{x} \vee \bar{y} \vee z)(x \vee \bar{y} \vee z)(x \vee y \vee z).$$

Приклад 7.4. Записати у досконалій нормальній формі булеву функцію $y(x_1, x_2, x_3)$, яка задана таблицею:

x_1	0	0	0	0	1	1	1	1
x_2	0	0	1	1	0	0	1	1
x_3	0	1	0	1	0	1	0	1
y	0	1	1	0	1	0	0	1

$$y = \bar{x}_1\bar{x}_2x_3 \vee \bar{x}_1x_2\bar{x}_3 \vee x_1\bar{x}_2\bar{x}_3 \vee x_1x_2x_3 =$$
$$= (x_1 \vee x_2 \vee x_3)(x_1 \vee \bar{x}_2 \vee \bar{x}_3)(\bar{x}_1 \vee x_2 \vee \bar{x}_3)(\bar{x}_1 \vee \bar{x}_2 \vee x_3).$$

7.3. Завдання для самостійного виконання.

Записати логічні вирази в диз\`юнктивній та досконалій диз\`юнктивній формі.

1. $a\bar{b} \vee \bar{b}c \vee (a \to bc)$.
2. $(\bar{a} \vee \bar{b} \vee \bar{c})\cdot(ab \vee c)$.
3. $(a \leftrightarrow \bar{b}) \vee (ac + (b \to c))$.
4. $\bar{a} \cdot b \vee \bar{c}$.
5. $(\bar{a}b + c)\cdot(ac \to b)$.
6. $(a \leftrightarrow b) \wedge (a\bar{b} \vee \overline{ab})$.
7. $\overline{a\bar{b}} \leftrightarrow \bar{a} \vee a \wedge b$.
8. $\bar{a}\cdot\bar{b} \vee b\cdot\bar{c} \vee a\cdot\bar{c}$.
9. $(a \to b) + (a|b\cdot c)$.
10. $(a \to b) \wedge (c \to b)$.
11. $(\bar{a}b \to c)\cdot((a \to c) \to b)$.
12. $a \wedge b \to (a \to \bar{b})$.
13. $\overline{(a \vee \bar{b})}(\overline{a \wedge b})$.
14. $a \wedge b \to (\bar{b} \wedge b \to c)$.
15. $(a \vee b)\cdot(\bar{b} \vee c)\cdot\bar{c}$.
16. $(a \wedge (a \vee b)) \wedge (\bar{b} \to a)$.
17. $a \vee b\cdot c \vee \bar{b}\cdot\bar{c}$.
18. $((a \to b) \wedge c) \vee \bar{a} \wedge b$.
19. $\bar{a}\cdot(\bar{b} \vee c)$.
20. $a \vee \bar{b} \to c \wedge b$.
21. $a \vee b \to (a \leftrightarrow b)$.
22. $(a \to b) \leftrightarrow (\bar{a} \vee b)$.
23. $(\bar{a} \vee b)\cdot(a \vee \bar{c})\cdot(b \vee c)$.
24. $(a\bar{b})(\overline{a \vee b})$.
25. $\overline{a \wedge \bar{b}} \leftrightarrow \bar{a}$.
26. $(a \to b) \to (b \to a)$.
27. $a \vee b \to (a \to \bar{b})$.
28. $\overline{(a \wedge b) \vee c}$.
29. $a \vee b \leftrightarrow \bar{a}$.
30. $(a \leftrightarrow b) \to \bar{c}$.

ТЕМА 8.

МІНІМІЗАЦІЯ БУЛЕВИХ ФУНКЦІЙ МЕТОДОМ КВАЙНА.

Ключові слова: імпліканта, мінімальна нормальна форма, склеювання, метод Квайна.

8.1. Основні теоретичні відомості.

Нехай булева функція приведена до ДДНФ. Тоді, використовуючи закони поглинання та склеювання, можливе *спрощення* булевої функції.

Як вже розглядалось, повний набір термів утворює конституенту. У процесі ж спрощення деякі терми з конституент зникають. При цьому, можливе розмноження конституент за допомогою закону ідемпотентності. Частину кон'юнкта, що залишився, називають *імплікантой*. Імпліканти з'являються в результаті склейки *суміжних* конституент, що різняться одним термом. Однак для функцій, що залежать від багатьох змінних, неконтрольований процес склейки неминуче приводить до зайвих імплікант, видалення яких не змінить значення функції на відповідних наборах змінних. Необхідне число імплікант може виявитися набагато менше можливого числа суміжних склейок. Методи, які дозволяють одержати найбільш скорочену (мінімальну) нормальну форму булевої функції, називають методами мінімізації булевих функцій.

Застосування цих методів дозволяє знайти більш компактні аналітичні вирази для заданої функції, тобто *мінімальну диз'юнктивну нормальну форму*, або *мінімальну кон’юнктивну нормальну форму*.

Зауважимо, що розглянуті далі методи мінімізації стосуються тільки випадку, коли булева функція задано у ДДНФ. Ці методи на підставі принципу подвійності легко можуть бути поширені й на ДКНФ.

Розглянемо метод отримання скороченої ДНФ, який називається *методом Квайна*. Задача мінімізації за методом Квайна полягає в попарному порівнянні всіх імплікант, що входять до ДДНФ, з метою виявлення можливості поглинання будь-якої змінної. Для цього складається таблиця Квайна, в рядках якої записуються знайдені первинні імпліканти, а в стовпцях вказуються конституенти ДДНФ функції. Клітинки цієї таблиці відмічаються у випадку, коли первинна імпліканта входить до складу конституенти. Після цього, задача спрощення зводиться до того, щоб знайти таку мінімальну кількість первинних імплікант, які покривають всі стовпці таблиці Квайна.

8.2. Приклади виконання завдань.

Приклад 8.1. Знайти мінімальну ДНФ функції $f = \overline{x\overline{(y \vee z)} \wedge (x \vee yz)}$.

$$f = x\overline{\overline{(y \vee z)} \wedge (x \vee yz)} = x(y \vee z) \vee \overline{(x \vee yz)} = x(y \vee z) \vee \bar{x}\,\overline{(yz)} =$$
$$= x(y \vee z) \vee \bar{x}(\bar{y} \vee \bar{z}) = xy \vee xz \vee \bar{x}\bar{y} \vee \bar{x}\bar{z}.$$

Застосувавши розгортку, знаходимо ДДНФ:

$$xy(\bar{z} \vee z) \vee xz(\bar{y} \vee y) \vee \bar{x}\bar{y}(\bar{z} \vee z) \vee \bar{x}\bar{z}(\bar{y} \vee y) =$$
$$= xyz \vee \bar{x}\bar{y}z \vee xyz \vee \bar{x}\bar{y}z \vee \bar{x}\bar{y}z \vee \bar{x}\bar{y}\bar{z} \vee x\bar{y}z \vee xy\bar{z} =$$
$$= xyz \vee x\bar{y}z \vee \bar{x}\bar{y}z \vee \bar{x}\bar{y}z \vee \bar{x}\bar{y}\bar{z} \vee x\bar{y}\bar{z}.$$

Для пошуку мінімальної нормальної форми функції складемо таблицю Квайна:

	xyz	xyz	xyz	$x\ yz$	$x\ y\ z$	xyz
xy	$\vee$	$\vee$				
xz	$\vee$		$\vee$			
$\bar{x}\bar{y}$				$\vee$	$\vee$	
$\bar{x}\bar{z}$					$\vee$	$\vee$

Після заповнення таблиці Квайна треба виконати покриття імплікантами стовпців таблиці. Для цього треба визначитися з обов'язковими імплікантами, які складають ядро покриття. Імпліканта називається обов'язковою, якщо знайдеться конституента, яка належить даній імпліканті і тільки їй.

У даному випадку всі імпліканти є обов'язковими. Отже, мінімальна нормальна форма функції наступна: $f = xy \vee xz \vee \bar{x}\bar{y} \vee \bar{x}\bar{z}$.

Приклад 8.2. Іноді, для більш раціонального запису логічної функції використовують десяткові еквіваленти (номери) наборів змінних, на яких функція приймає одиничне (нульове) значення. Наприклад, нехай задана булева функція чотирьох змінних: $f(x_1, x_2, x_3)\,|_1 = \vee(2,5,6,7,10,12,13,14)$.

Зрозуміло, що знак диз'юнкції у правій частині означення функції відповідає досконалій диз'юнктивній нормальній формі функції.

Запишемо ДДНФ функції:

$$f\,(0010, 0101, 0110, 0111, 1010, 1100, 1101, 1110) = \bar{x}_4\bar{x}_3x_2\bar{x}_1 \vee \bar{x}_4x_3\bar{x}_2\bar{x}_1 \vee$$
$$\vee\ \bar{x}_4x_3x_2\,\bar{x}_1 \vee \bar{x}_4\,x_3x_2\,x_1 \vee x_4\bar{x}_3x_2\bar{x}_1 \vee x_4\,\bar{x}_3\bar{x}_2\,x_1 \vee x_4\,\bar{x}_3x_2x_1 \vee x_4\,x_3\bar{x}_2\,x_1.$$

На першому етапі мінімізації вихідну ДДНФ можна спростити за рахунок використання закону склеювання, тоді одержимо:

$$f = x_2\bar{x}_1 \vee \bar{x}_4x_3x_1 \vee \bar{x}_4x_3x_2 \vee x_3\bar{x}_2x_1 \vee x_4x_3\bar{x}_2 \vee x_4x_3\bar{x}_1.$$

Звертаємо увагу на те, що ту саму конституенту (імпліканту) можна склеювати з іншими конституентами (імплікантами) багаторазово.

На другому етапі будуємо таблицю Квайна:

$x_4x_3x_2x_1$	0010	0101	0110	0111	1010	1100	1101	1110
- - 10	∨		∨		∨			∨
01 - 1		∨		∨				
011 -			∨	∨				
- 101		∨					∨	
110 -						∨	∨	
11- 0						∨		∨

Ядром покриття стовпців таблиці є перша імпліканта (- -10), тобто, ця імпліканта обов'язкова. Вона покриває 4 стовпця. Далі треба вибрати такі імпліканти, які разом з обов'язковою імплікантою покривають усі стовпці. Такими будуть друга та п'ята імпліканти. Виявилося, що можна обійтися тільки трьома імплікантами замість шести. Отже, мінімальна нормальна форма функції наступна: $f = x_2\bar{x}_1 \vee \bar{x}_4x_3x_1 \vee x_4x_3\bar{x}_2$.

За допомогою таблиць істинності легко перевірити, що отримана в МНФ функція відтворює всі значення вихідної функції.

8.3. Завдання для самостійного виконання.

Нехай булева функція чотирьох змінних задана номерами наборів змінних, на яких вона приймає значення 1 (табл. 8.1). Запишіть цю функцію в ДДНФ й виконайте її мінімізацію методом Квайна.

Таблиця 8.1

№	Номери конституент							
1	4	6	8	9	10	11	15	-
2	2	3	6	7	8	14	15	-
3	0	2	4	5	6	7	9	11
4	1	3	5	7	8	12	14	-
5	0	3	7	9	10	12	13	14
6	1	2	5	6	10	12	13	14
7	0	2	5	8	10	11	14	15
8	0	1	2	4	7	10	11	12
9	0	5	7	8	9	12	13	15
10	0	1	2	3	9	12	14	15
11	0	1	4	6	7	8	9	15
12	0	3	4	5	7	8	10	11
13	0	2	3	7	8	12	14	15
14	0	2	9	10	11	12	14	15
15	1	2	5	6	8	9	10	14
16	1	3	6	7	9	11	13	-
17	1	6	7	9	12	13	14	15
18	1	2	4	10	11	13	14	-
19	1	5	6	7	9	13	14	15
20	1	2	3	4	9	12	15	-
21	2	3	4	7	10	11	12	14
22	2	3	5	8	10	11	12	14
23	3	4	5	7	8	9	10	11
24	4	5	6	9	10	11	12	15
25	2	3	6	7	10	11	13	14
26	2	4	5	8	9	11	12	14
27	3	4	5	7	8	9	13	14
28	3	4	6	7	8	11	12	13
29	4	5	8	9	10	11	14	15
30	4	5	6	7	10	11	13	15

ТЕМА 9.

МІНІМІЗАЦІЯ БУЛЕВИХ ФУНКЦІЙ МЕТОДОМ СПОЛУЧЕННЯ ІНДЕКСІВ.

Ключові слова: індекс, комбінація індексів, коди індексів, метод сполучення індексів.

9.1. Основні теоретичні відомості.

Не менш ефективним способом мінімізації логічних функцій є *метод сполучення індексів*. В основі метода лежить використання таблиці індексів (табл. 9.1). Під поняттям «індекси», мається на увазі, саме, індекси змінних.

Таблиця індексів складена з наступних частин. Перша частина – це перші чотири стовпця, які позначені, як 1, 2, 3, 4 – це індекси. Кожному рядку цієї частини відповідає двоїчний еквівалент номеру рядка. Отже, кожному індексу у стовпці виписано коди (0 або 1).

Наступна частина таблиці основна. Стовпці цієї частини таблиці позначені як можливі комбінації індексів (без урахування порядку). В кожному рядку цієї частини відповідно до комбінації індексів виписано комбінації кодів цих індексів з того ж рядка кодів індексів першої частини таблиці. Наприклад, в рядку під номером 7 та стовпці під номером 124 записано код 110, тому що в цьому рядку індексу 1 відповідає код 1, індексу 2 – код 1, та індексу 4 – код 0.

В останньому стовпці таблиці виписані значення функції, при цьому значення змінних відповідають кодам першої частини таблиці.

Отже, для булевих функцій від чотирьох змінних, крім останнього стовпця, таблиця одна ж сама.

Подальший аналіз цього методу краще провести на конкретному прикладі. Скористаємось логічною функцією попередньої теми:

$$f = \bar{x}_4\bar{x}_3x_2\bar{x}_1 \vee \bar{x}_4x_3\bar{x}_2\bar{x}_1 \vee$$
$$\vee \bar{x}_4x_3x_2\bar{x}_1 \vee \bar{x}_4x_3x_2x_1 \vee \bar{x}_4x_3\bar{x}_2x_1 \vee x_4\bar{x}_3\bar{x}_2x_1 \vee x_4\bar{x}_3x_2x_1 \vee x_4x_3\bar{x}_2x_1.$$

Аналіз таблиці починається ліворуч по стовпцях. Принцип виключення кодів наступний. Усі коди в рядках, що закінчуються нульовим значенням функції, виключаються автоматично. Якщо ці коди попадають на рядки, що закінчуються значенням функції 1, то вони також не враховуються. Залишаються тільки ті коди, які розташовані на рядках з одиничним значенням функції (ці коди затемнені).

Таблиця 9.1

n	**1**	**2**	**3**	**4**	**12**	**13**	**14**	**23**	**24**	**34**	**123**	**124**	**134**	**234**	**1234**	***y***
0	0	0	0	0	00	00	00	00	00	00	000	000	000	000	0000	**0**
1	1	0	0	0	10	10	10	00	00	00	100	100	100	000	1000	**0**
2	0	1	0	0	01	00	00	10	10	00	010	010	000	100	0100	**1**
3	1	1	0	0	11	10	10	10	10	00	110	110	100	100	1100	**0**
4	0	0	1	0	00	01	00	01	00	10	001	000	010	010	0010	**0**
5	1	0	1	0	10	11	10	01	00	10	101	100	110	010	1010	**1**
6	0	1	1	0	01	01	00	11	10	10	011	010	010	110	0110	**1**
7	1	1	1	0	11	11	10	11	10	10	111	110	110	110	1110	**1**
8	0	0	0	1	00	00	01	00	01	01	000	001	001	001	0001	**0**
9	1	0	0	1	10	10	11	00	01	01	100	101	101	001	1001	**0**
10	0	1	0	1	01	00	01	10	11	01	010	011	001	101	0101	**1**
11	1	1	0	1	11	10	11	10	11	01	110	111	101	101	1101	**0**
12	0	0	1	1	00	01	01	01	01	11	001	001	011	011	0011	**1**
13	1	0	1	1	10	11	11	01	01	11	101	101	111	011	1011	**1**
14	0	1	1	1	01	01	01	11	11	11	011	011	011	111	0111	**1**
15	1	1	1	1	11	11	11	11	11	11	111	111	111	111	1111	**0**

Далі керуються наступним правилом. Для того, щоб функція прийняла значення, рівне одиниці, досить того, щоб тільки яка-небудь одна імпліканта на рядку прийняла одиничне значення. Насамперед, залишаємо код 01 в рядку «2» та стовпці «12», який відповідає імпліканті $x_2\bar{x}_1$. Цей код перекриває одиниці в рядках 2, 6, 10 і 14. Потім, природно, звертаємося до 12–го рядка. Тут залишаємо єдиний на рядку код 011, що відповідає імпліканті $x_4x_3\bar{x}_2$. Ця ж імпліканта відповідальна за 13–ий рядок, що кінчається теж одиницею. Залишилося розглянути 5-ий й 7-ий рядки. Загальної для них є імпліканта $\bar{x}_4x_3x_1$. Отже, знову маємо: $f = x_2\bar{x}_1 \vee \bar{x}_4x_3x_1 \vee x_4x_3\bar{x}_2$.

Хоча табл. 9.1 більш громіздка, ніж таблиця Квайна, метод сполучення індексів не вважається більш складним, ніж метод Квайна, якщо пам'ятати, що для складання таблиці Квайна необхідно спочатку зробити численні склейки конституент і імплікант. Реалізація на комп'ютері алгоритму методу сполучення індексів виявляється порівняно простою.

9.2. Завдання для самостійного виконання.

Нехай булева функція чотирьох змінних задана номерами наборів змінних, на яких вона приймає значення 1 (табл. 9.2). Виконайте її мінімізацію методом сполучення індексів.

Таблиця 9.2

№	Номери конституент							
1	3	5	7	10	11	13	14	15
2	1	4	5	7	9	10	11	15
3	1	2	3	5	7	8	9	12
4	0	2	4	5	8	10	12	13
5	0	1	6	8	11	12	13	15
6	1	3	5	7	8	9	11	12
7	1	2	3	7	8	10	12	13
8	0	1	2	3	5	9	10	12
9	1	3	4	8	9	10	13	15
10	1	5	6	7	9	12	13	15
11	0	1	2	6	7	9	10	12
12	1	3	4	5	8	9	11	12
13	0	1	2	7	8	9	14	15
14	0	2	5	6	7	12	14	15
15	1	4	5	6	7	9	13	14
16	0	2	4	6	8	10	12	-
17	1	3	5	7	9	11	13	15
18	1	3	4	7	8	9	14	-
19	0	3	5	6	7	9	14	15
20	1	2	5	6	9	13	14	15
21	2	4	5	9	10	11	12	14
22	2	3	5	8	11	12	14	15
23	3	5	6	7	8	9	10	11
24	2	4	5	7	9	10	11	13
25	1	3	5	9	10	11	13	14
26	1	2	5	7	8	11	12	15
27	2	4	5	6	8	9	13	14
28	3	5	6	7	8	10	11	13
29	2	4	6	9	10	11	12	15
30	1	3	6	7	9	11	13	14

ТЕМА 10.

АЛГЕБРА ЖЕГАЛКІНА.

Ключові слова: алгебра Жегалкіна, поліном Жегалкіна.

10.1. Основні теоретичні відомості.

Алгебра Жегалкіна будується на основі операцій додавання по модулю 2 і кон'юнкції.

Безпосередньою перевіркою за таблицями істинності встановлюються наступні основні властивості цієї алгебри:

– комутативність $x + y = y + x$; $xy = yx$;

– асоціативність $x + (y + z) = (x + y) + z$; $x(yz) = (xy)z$;

– дистрибутивність множення щодо додавання $x(y + z) = xy + xz$;

– властивості констант $x\cdot 1 = x$; $x \cdot 0 = 0$; $x + 0 = x$.

Усі ці властивості подібні властивостям звичайної алгебри, але на відміну від булевої алгебри, закон дистрибутивності додавання щодо множення не має сили $x + yz$ □ $(x+ y)(x+ z)$.

Мають місце також наступні тотожності:

– закон приведення подібних членів при додаванні $x + x = 0$;

– закон ідемпотентності для множення $xx = x$.

Таким чином, у формулах алгебри Жегалкіна, як і в булевій алгебрі, не можуть з'являтися коефіцієнти при змінних і показники ступеня.

Перейти від будь-якої формули булевої алгебри до відповідної її формули алгебри Жегалкіна можна за допомогою наступних тотожностей:

$$\bar{x} = 1 + x;\ x_1 \vee x_2 = x_1 + x_2 + x_1 x_2 ,$$

а зворотний перехід здійснюється з використанням тотожності:

$$x_1 + x_2 = x_1 \bar{x}_2 \vee \bar{x}_1 x_2.$$

Через операції алгебри Жегалкіна можна виразити всі інші булеві функції:

$x_1 \to x_2 = \bar{x}_1 \vee x_2 = 1 + x_1 + x_1 x_2;$

$x_1 \leftrightarrow x_2 = (\bar{x}_1 \vee x_2)(x_1 \vee \bar{x}_2) = 1 + x_1 + x_2;$

$x_1 \leftarrow x_2 = \overline{x_1 \to x_2} = x_1 + x_1 x_2;$

$x_1 \mid x_2 = \overline{x_1 x_2} = 1 + x_1 x_2;$

$x_1 \downarrow x_2 = \overline{x_1 \vee x_2} = 1 + x_1 + x_2 + x_1 x_2.$

Перевага алгебри Жегалкіна полягає в арифметизації логіки, що дозволяє виконувати перетворення булевих функцій, використовуючи досвід перетворення звичайних алгебраїчних виразів. Її недолік у порівнянні з булевою алгеброю – складність формул, що особливо позначається при значному числі змінних.

10.2. Приклади виконання завдань.

Приклад 10.1. Побудувати поліном Жегалкіна.

$$x(x \vee \bar{y}) = x[(1+x) + y + (1+x)y] = x(1+x+y+y+xy) =$$
$$= x(1+x+xy) = x + xx + xxy = x + x + xy = xy.$$

Приклад 10.2. Записати поліном Жегалкіна формулою алгебри Буля.

$$1 + x + y + xy = (1+x)(1+y) = \bar{x}\bar{y}\ .$$

Приклад 10.3. Побудуємо поліном Жегалкіна для функції $x \leftrightarrow y$.

1 спосіб. Використаємо метод невизначених коефіцієнтів. Записуємо поліном Жегалкіна з невідомими коефіцієнтами a_0, a_1, a_2, a_{12} для функції від двох змінних $x_1 \leftrightarrow x_2 = a_0 + a_1 x_1 + a_2 x_2 + a_{12} x_1 x_2$.

Будуємо таблицю істинності для $x_1 \leftrightarrow x_2$:

x_1	x_2	$x_1 \leftrightarrow x_2$
0	0	1
0	1	0
1	0	0
1	1	1

Послідовно виконуємо підстановку значень x_1 й x_2 з таблиці у вираз $a_0 + a_1 x_1 + a_2 x_2 + a_{12} x_1 x_2$ для визначення невідомих коефіцієнтів.

$x_1=0$, $x_2=0$: $1 = a_0$;

x_1=0, x_2=1: 0=1+ a_2 $\Rightarrow$ a_2=1;

x_1=1, x_2=0: 0=1+ a_1 $\Rightarrow$ a_1=1;

x_1=1, x_2=1: 1=1+ a_{12} $\Rightarrow$ a_{12}=0.

Поліном Жегалкіна для еквівалентності має вид: $x_1 \leftrightarrow x_2 = 1 + x_1 + x_2$.

2 спосіб. Розглянемо метод еквівалентних перетворень. Спочатку запишемо ДДНФ даної логічної функції: $x_1 \leftrightarrow x_2 = \bar{x}_1\bar{x}_2 \vee x_1x_2$. Так як, $x \vee y = x + y + xy$, то $x_1 \leftrightarrow x_2 = \bar{x}_1\bar{x}_2 \vee x_1x_2 = \bar{x}_1\bar{x}_2 + x_1x_2 + \bar{x}_1\bar{x}_2\, x_1x_2$.
Оскільки, $\bar{x}_1\bar{x}_2x_1x_2 = 0$ та $\bar{x} = 1 + x$, то $\bar{x}_1\bar{x}_2 + x_1x_2 + \bar{x}_1\bar{x}_2\, x_1x_2 = \bar{x}_1\bar{x}_2 + x_1x_2$.

$x_1 \leftrightarrow x_2 = (1 + x_1)(1 + x_2) + x_1x_2 = 1 + x_1 + x_2 + x_1x_2 + x_1x_2 = 1 + x_1 + x_2$.

Приклад 10.4. Побудувати поліном Жегалкіна для функції $\overline{(x_1 \wedge x_2) \vee x_3}$.

Поліном з невідомими коефіцієнтами має вид:

$$f = a_0 + a_1x_1 + a_2x_2 + a_3x_3 + a_{12}x_1x_2 + a_{13}x_1x_3 + a_{23}x_2x_3 + a_4x_1x_2x_3.$$

Використаємо метод невизначених коефіцієнтів. Побудуємо таблицю істинності.

x_1	x_2	x_3	$x_1 \wedge x_2$	$(x_1 \wedge x_2) \vee x_3$	$\overline{(x_1 \wedge x_2) \vee x_3}$
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	0	1	0
1	0	0	0	0	1
1	0	1	0	1	0
1	1	0	1	1	0
1	1	1	1	1	0

Для наочності обчислень невідомих коефіцієнтів можна скористатися таблицею. В формулу полінома послідовно виконуємо підстановку значень змінних x_1, x_2, x_3.

x_1	x_2	x_3	f	
0	0	0	1	1= a_0
0	0	1	0	0=1+ a_3, $\Rightarrow$ a_3 =1
0	1	0	1	1=1+ a_2, $\Rightarrow$ a_2 =0

x_1	x_2	x_3	f	
0	1	1	0	0=1+1+ a_{23}, $\Rightarrow a_{23}$ =0
1	0	0	1	1=1+ a_1, $\Rightarrow a_1$ =0
1	0	1	0	0=1+1+ a_{13}, $\Rightarrow a_{13}$ =0
1	1	0	0	0=1+ a_{12}, $\Rightarrow a_{12}$ =1
1	1	1	0	0=1+1+1+ a_4, $\Rightarrow a_4$ =1

Отже, $\overline{(x_1 \wedge x_2) \vee x_3} = 1 + x_3 + x_1 x_2 + x_1 x_2 x_3$.

10.3. Завдання для самостійного виконання.

Побудувати поліном Жегалкіна для наступних логічних функцій.

1. $x \to (y \to \bar{x} \cdot y)$.

2. $(x \leftrightarrow y) \wedge (y \leftrightarrow z)$.

3. $(x \to \bar{y}) \vee (\overline{x \vee y})$.

4. $(x \vee y) \cdot (|x\, z \to y)$.

5. $f = (10101100)$.

6. $(x \to y) \leftrightarrow (z \to (x \leftrightarrow \bar{z}))$.

7. $x \cdot (y \leftrightarrow x \cdot \bar{y})$.

8. $(x \to (y \to z)) \cdot ((x \to y) \to z)$.

9. $f = (11000100)$.

10. $(x \vee y \cdot \bar{z}) + (x \cdot \bar{y} \leftrightarrow (x \to y \cdot \bar{z}))$.

11. $\overline{(x \wedge y)(x \to y)}$.

12. $x \wedge \bar{y} \to (\bar{y} \to x)$.

13. $f = (0010110)1$.

14. $x \vee \bar{y} \vee ((x + (y \downarrow z)) \overline{\to} y \cdot z)$.

15. $\overline{(\bar{x} \to y)}$.

16. $((x \leftrightarrow y) \downarrow z) \cdot (\bar{y} \cdot z \to (x + y \cdot z))$.

17. $(x \vee y) \wedge \bar{x} \to y$.

18. $(x \vee y) \leftrightarrow (\bar{x} \to \bar{z})$.

19. $(x \downarrow y) | z$.

20. $x \downarrow ((x \to y) \vee \bar{z})$.

21. $(x \vee y)\overline{(x \leftrightarrow y)}$.

22. $(x \vee y \vee \bar{z}) \to (x \cdot \bar{y} \leftrightarrow x + y \cdot \bar{z})$.

23. $(x \downarrow y) | (y \downarrow z)$.

24. $\overline{(x \leftrightarrow y)}(y \leftrightarrow z)$.

25. $x \leftrightarrow y \leftrightarrow \bar{z}$.

26. $\overline{(x \to y)} \leftrightarrow (z \vee (x \leftrightarrow \bar{z}))$.

27. $\overline{x \to y \to \bar{x} \to \bar{z}}$.

28. $(x \vee y) \to (\bar{x} \vee z)$.

29. $(x \leftrightarrow y)\overline{(\bar{y} \leftrightarrow z)}$.

30. $\overline{(x \leftrightarrow y) \wedge (y \leftrightarrow z)}$.

ТЕМА 11.

КЛАСИ БУЛЕВИХ ФУНКЦІЙ.

Ключові слова: монотонна функція, самодвоїста функція, лінійна функція, функціональна повнота.

11.1. Основні теоретичні відомості.

Із множини $\nu = 2^{2^n}$ різних булевих функцій *n* змінних виділяються наступні п'ять типів булевих функцій.

1) *Функція, що зберігає константу* 0, тобто $f(0, 0, ..., 0) = 0$.

Так як на одному з 2^n наборів $(x_1, x_2, ..., x_n)$ значення таких функцій фіксовані, то їх число рівне $2^{2^{n-1}} = \frac{1}{2}2^{2^n} = \frac{1}{2}\nu$, тобто половина всіх функцій зберігає константу 0.

2) *Функція, що зберігає константу* 1, тобто $f(1, 1, ..., 1) = 1$.

Число таких функцій, як і в попередньому випадку, дорівнює іншій половині загального числа всіх функцій.

3) *Самодвоїста функцяї* – це така булева функція, яка приймає протилежні значення на будь-яких двох протилежних наборах змінних.

Самодвоїста функція повністю визначається завданням її значень на половині всіх наборів (інші значення визначаються за умовою анти симетричності), тому число незалежних наборів рівно $\frac{1}{2}2^n$ й число всіх таких функцій $2^{\frac{1}{2}2^n} = \sqrt{2^{2^n}} = \sqrt{\nu}$.

4) *Лінійна функція* – це така функція, яка представляється в алгебрі Жегалкіна канонічним многочленом що не містить добутків змінних. У такого многочлена $a_0 + a_1x_1 + ... + a_nx_n$, де коефіцієнти $a_0, a_1, ..., a_n$ приймають значення 0 або 1, всього коефіцієнтів $n + 1$. Отже число різних лінійних многочленів буде 2^{n+1}. В силу однозначності зображення функції канонічним многочленом це число виражає й кількість лінійних функцій.

5) *Монотонна функція* – це така функція *f*, яка для будь-яких двох частково впорядкованих наборів змінних $\alpha = (\alpha_1, \alpha_2, ..., \alpha_n) \leq \beta = (\beta_1, \beta_2, ..., \beta_n)$, де $\alpha_i \leq \beta_i$ $(i = 1, 2, ..., n)$, задовольняє нерівності $f(\alpha) \leq f(\beta)$.

Розглянуті типи функцій замкнені щодо операції суперпозиції, тобто суперпозиція будь-якого числа булевих функцій даного типу є функцією того ж

типу. Наступна табл. 11.1 містить властивості елементарних булевих функцій від двох змінних.

Таблиця 11.1

Позначення функції	Примітка	Зберіг. 0	Зберіг. 1	Само-двоїста	Моно-тонна	Лінійна
$y_0 = 0$	Т	□			□	□
$y_1 = x_1 \wedge x_2$		□	□		□	
$y_2 = x_1 - x_2$		□				
$y_3 = x_1$	Т	□	□	□	□	□
$y_4 = x_2 - x_1$		□				
$y_5 = x_2$	Т	□	□	□	□	□
$y_6 = x_1 + x_2$		□				□
$y_7 = x_1 \vee x_2$		□	□		□	
$y_8 = x_1 \downarrow x_2$	У					
$y_9 = x_1 \leftrightarrow x_2$			□			□
$y_{10} = \bar{x}_2 = \rceil x_2$				□		□
$y_{11} = x_1 \leftarrow x_2$			□			
$y_{12} = \bar{x}_1 = \rceil x_1$				□		□
$y_{13} = x_1 \rightarrow x_2$			□			
$y_{14} = x_1 \mid x_2$	У					
$y_{15} = 1$	Т		□		□	□

Примітка: Т – тривіальна; У – універсальна.

Система функцій, суперпозицією яких може бути представлена будь-яка функція з деякої множини булевих функцій, називається *функціонально повною*. Якщо в такій системі допускаються константи 0 і 1, то її називають *ослаблено функціонально повною*.

Загальне вирішення щодо *функціональної повноти* наступне: для того, щоб система булевих функцій була повною, необхідно й достатньо, щоб вона включала хоча би одну функцію, яка не зберігає константу 0, не зберігає константу 1, не самодвоїсту, не лінійну й не монотонну.

З табл. 11.1 видно, що такі функції, така як стрілка Пірса та штрих Шеффера, задовольняють умовам функціональної повноти, тобто для запису будь-якої булевої функції досить тільки однієї з двох цих функцій Отже, ці функції є універсальними. До речі, є корисними співвідношення:

$$\bar{x} = x \downarrow x = x \mid x\ ;$$
$$x_1 x_2 = (x_1 \mid x_2) \mid (x_1 \mid x_2)\ ;$$
$$x_1 \vee x_2 = (x_1 \downarrow x_2) \downarrow (x_1 \downarrow x_2).$$

Системи операцій: диз'юнкція й заперечення, кон'юнкція й заперечення, задовольняють теоремі про функціональну повноту.

Система операцій алгебри Жегалкіна (сума по модулю 2 і кон'юнкція) разом з константою 1 утворює ослаблено функціонально повну систему.

Вибравши будь-яку елементарну функцію й доповнивши її однієї або декількома іншими функціями так, щоб усі вони разом задовольняли твердженню про функціональну повноту, можна виразити через них усі інші булеві функції.

Наприклад, в основу одного з таких комплектів можна покласти імплікацію й константу 0: $x_1 \vee x_2 = (x_1 \to x_2) \to x_2$ і $\bar{x} = x \to 0$.

Інший приклад функціонально повного комплекту є кон'юнкція, еквівалентність й константа 0. При цьому, формули алгебри логіки, побудованої на цих операціях, будуть двоїсті формулам алгебри Жегалкіна, якщо в якості двоїстих символів прийняти «+» і «↔», а також 1 і 0.

Очевидно, усе краще, що можна витягти з різних варіантів функціонально повних систем, уже закладене в булевої алгебрі й алгебрі Жегалкіна. Але, при розв'язку спеціальних задач, не виключається побудова й застосування інших алгебр логіки.

11.2. Приклади виконання завдань.

Приклад 11.1. Перевірити булеву функцію $f = (x_1 \to x_2) \to x_1 x_3$ на самодвоїстість.

Запишемо її в приведеному виді:

$$(x_1 \to x_2) \to x_1 x_3 = \overline{(\bar{x}_1 \vee x_2)} \vee x_1 x_3 = x_1 \bar{x}_2 \vee x_1 x_3 = x_1(\bar{x}_2 \vee x_3).$$
$$f(x_1, x_2, x_3) = x_1(\bar{x}_2 \vee x_3).$$

Далі, згідно з означенням самодвоїстої функції, запишемо:

$$\overline{f(\bar{x}_1, \bar{x}_2, \bar{x}_3)} = \overline{\bar{x}_1(x_2 \vee \bar{x}_3)} = x_1 \vee \overline{(x_2 \vee \bar{x}_3)} = x_1 \vee \overline{x_2}\, x_3 .$$

Нехай, $x_1 = 0$, $x_2 = 0$, $x_3 = 1$, тоді $f(x_1, x_2, x_3) = 0$, $\bar{f}(\bar{x}_1, \bar{x}_2, \bar{x}_3) = 1$.
Тому $f(x_1, x_2, x_3) \neq \overline{f(\bar{x}_1, \bar{x}_2, \bar{x}_3)}$, що означає, що функція не самодвоїста.

Приклад 11.2. Перевірити функцію $f = (1011)$ на монотонність.

Функція $f = (1011)$ не монотонна, так як для $(00) < (01)$ маємо $f(0,0) > f(0,1)$.

Приклад 11.3. Перевірити систему $\{x_1 \leftrightarrow x_2, \overline{x_1}, \overline{x_1} \to \overline{x_2}\}$ на повноту.

Для перевірки системи на повноту треба перевірити, що система має:
– функцію, яка не зберігає 0;
– функцію, яка не зберігає 1;
– не монотонну функцію;
– не самодвоїсту функцію;
– не лінійну функцію.

1. Очевидно, що функція $f_1(x_1, x_2) = x_1 \leftrightarrow x_2$ не зберігає 0, так як $f_1(0,0) = 1$.

2. Перевіримо, чи є f_1 самодвоїстою.

$\bar{x}_1$	$\bar{x}_2$	$\bar{x}_1 \leftrightarrow \bar{x}_2$	$\overline{f_1(\bar{x}_1, \bar{x}_2)}$
1	1	1	0
1	0	0	1
0	1	0	1
0	0	1	0

Так як $f_1(x_1, x_2) \neq \overline{f_1(\bar{x}_1, \bar{x}_2)}$, то f_1 не самодвоїста.

3. Функція $f_2(x) = \bar{x}$ не монотонна та не зберігає 1.

4. Знайдемо поліном Жегалкіна для функції:

$$f_3(x_1, x_2) = \bar{x}_1 \to \bar{x}_2 = a_0 + a_1 x_1 + a_2 x_2 + a_{12} x_1 x_2 .$$

x_1	x_2	$\bar{x}_1$	$\bar{x}_2$	$\bar{x}_1 \to \bar{x}_2$
0	0	1	1	1
0	1	1	0	0
1	0	0	1	1
1	1	0	0	1

$f_3(0, 0) = 1 \Rightarrow a_0 = 1;$

$f_3(0,1) = 0 \Rightarrow 1 + a_2 = 0 \Rightarrow a_2 = 1;$

$f_3(1, 0) = 1 \Rightarrow 1 + a_1 = 1 \Rightarrow a_1 = 0;$

$f_3(1,1) = 1 \Rightarrow 1 + 1 + a_{12} = 1 \Rightarrow a_{12} = 1.$

Функція $f_3(x_1, x_2) = \bar{x}_1 \to \bar{x}_2 = 1 + x_2 + x_1 x_2$ не лінійна.

Отже, система $\{x_1 \leftrightarrow x_2, \bar{x}_1, \bar{x}_1 \to \bar{x}_2\}$ повна.

11.3. Завдання для самостійного виконання.

1. Перевірити самодвоїстість функцій.

1. $x \wedge y \leftrightarrow (z \to x \vee \bar{y} \to y \vee x)$.

2. $\left(x \wedge (y + \bar{z}) \vee \bar{x} \wedge z\right) \to z \vee \bar{y}$.

3. $\bar{y} \wedge z + \left(\bar{x} \vee y \to y \vee \bar{z}\right)$.

4. (01101001).

5. $\overline{x \wedge y} \to (y \vee \overline{x \wedge z} \leftrightarrow \overline{x \vee y})$.

6. $\left(x \vee \bar{y} \vee \bar{z}\right) \to x \cdot \left(y \leftrightarrow z\right)$.

7. $x \wedge y \to (z \to x) \to y \vee \bar{x}$.

8. (10110110).

9. $\left(x \wedge y + \bar{z} \vee \bar{x} \wedge y\right) \to z$.

10. $\bar{y} \wedge x \to \left(\bar{x} \to y \vee \bar{x}\right)$.

11. (01010101).

12. $\overline{x \downarrow y} \vee x \to y$.

13. (10101010).

13. $\bar{x} \vee y \to x \wedge \bar{y} \to x \to y \vee x$.

15. $x \cdot y + y \cdot z$.

14. $\overline{x \to y} \vee \bar{z}$.

17. (10001110).

18. $x \vee y \leftrightarrow \bar{z} \wedge y \vee \bar{x}$.

19. $\overline{xy} \vee \bar{y} \vee z$.

20. (10101000).

2. Перевірити монотонність функцій.

1. $(x+y)\cdot(x \leftrightarrow y)$.
2. $\overline{x \to \bar{y}}$.
3. (00110111).
4. $x \cdot z(x+z)$.
5. $x \to (x \to y)$.
6. $(x \downarrow z) \to (y | \bar{z})$.
7. (01100111).
8. $\overline{\bar{x} \vee \bar{y}}$.
9. $\bar{x} \to yz$.
10. $(\bar{x} \to y) \vee xy$.
11. (10011100).
12. (00011100).
13. $\overline{(\bar{x} \vee \bar{y})}z$.
14. $(\bar{x} \to y) \vee z$.
15. (0010111).
16. (0011011).
17. $|\,(x \downarrow y) \to x + z$.
18. $(\bar{x} \leftrightarrow y) \wedge xy$.
19. $x \cdot y + y \cdot z$.
20. $(x+y)\cdot x \cdot y$.

3. Перевірити повноту наступних систем.

1. $\{x+y, \bar{x}\}$.
2. $\{\bar{x}+y, \bar{y}, x \vee \bar{y}\}$.
3. $\{x \to yz, x(y \leftrightarrow z), xy + yz\}$.
4. $\{x\bar{y}, \bar{x} \leftrightarrow x \cdot z\}$.
5. $\{x \to y, 0, x \leftrightarrow y\}$.
6. $\{0, 1, x(y \leftrightarrow x) \vee \bar{x}(x+z)\}$.
7. $\{x \to y, x \to \bar{y}z\}$.
8. $\{\bar{x}, (0010), (0100111001110001)\}$.
9. $\{xy \vee xz \vee yz, 0, 1\}$.
10. $\{(10), (00110111)\}$.
11. $\{xy \vee xz \vee yz, \bar{x}, \bar{x} \to y\}$.
12. $\{x \to \bar{y}, y, 0\}$.
13. $\{(11), (0111), (00110111)\}$.
14. $\{(1001), (11101000)\}$.
15. $\{x \leftrightarrow \bar{y}, x, \bar{x} \to \bar{y}\}$.
16. $\{\bar{x} \to y, y \to \bar{x}z\}$.
17. $\{\bar{x}, (0001), (0010101000100001)\}$.
18. $\{x \leftrightarrow xz, xz \to xy, x \vee y. x+y\}$.
19. $\{x \to y, \bar{x}\}$.
20. $\{xy, x \vee y, x \to y\}$.

ТЕМА 12.

ПОБУДОВА ОПТИМАЛЬНИХ КОДІВ МЕТОДОМ ФАНО.

Ключові слова: алфавіт, таблиця кодів, схема кодування, розділима схема, префіксна схема, оптимальне кодування, код Фано, вартість коду, кодове дерево.

12.1. Основні теоретичні відомості.

Предметом дослідження теорії кодування є відображення скінчених або лічених множин об'єктів довільної природи на множини послідовностей з цифр 0, 1, …, r–1, де r – деяке ціле додатне число. Такі відображення називаються *кодуваннями*.

Типова задача теорії кодування формулюється таким чином: при заданих алфавітах A, B та множині повідомлень S, знайти оптимальне кодування E, яке володіє певними властивостями.

Оптимальність, як правило, пов'язана з мінімізацією довжини кодів.

При кодуванні, повідомлення з множини S можна розглядати як єдине ціле і зіставляти йому якийсь код, або будувати код повідомлення з кодів його частин. Очевидно, елементарною частиною повідомлення є буква (літера) алфавіту A. Якщо задати таблицю кодів, тобто зіставити код кожній букві алфавіту, то будемо мати *схему алфавітного кодування*.

Якщо коди усіх букв алфавіту мають однакову довжину, то таку схему називають *схемою рівномірного кодування*.

Схема алфавітного кодування називається *розділимою*, якщо будь-яке кодове слово відповідного повідомлення єдиним чином розкладається на елементарні коди, тобто коди букв .

Очевидно, алфавітне кодування є взаємно однозначним тоді і тільки тоді, коли воно задається за допомогою розділимого коду. До речі, схема рівномірного кодування є розділимою.

Нехай задано алфавіт $A=\{a_1,\dots,a_n\}$, схема кодування $\sigma=\langle a_i \to \beta_i \rangle_{i=1}^{n}$ та ймовірності появи букв у повідомленні $P=\langle p_1,\dots, p_n \rangle$, де $p_1+\dots+p_n=1$. Тоді, число

$$l_\sigma(P)=\sum_{i=0}^{n} p_i l_i,$$

де $l_i=|\beta_i|$, називають *вартістю* кодування σ при розподілу P.

Теорема. Існує таке кодування σ *, що $l_{\sigma^*}(P) = \inf_{\sigma} l_{\sigma}(P)$.

Кодування σ * називають *оптимальним* для розподілу *P*.

Якщо $x = x_1x_2$, де x , x_1, x_2 слова деякого алфавіту *X*, то x_1 називають префіксом слова *x*.

Наступний алгоритм будує префіксну 68озділимо схему, близьку до оптимальної.

Алгоритм Фано.

1. Спочатку список букв повідомлення розташовуємо в порядку спадання їх ймовірностей.

2. Далі, список поділяється на дві частини так, щоб сума ймовірностей першої частини як можна менше відрізнялась від суми ймовірностей другої частини, тобто знаходиться медіана списку. Визначення медіани списку може бути неоднозначним.

3. Буквам з першої частини приписується символ «0», а буквам другої частини – символ «1».

4. Процедура повторюється для кожної частини окремо, якщо вона містить хоча б дві букви. Цей дихотомічний процес продовжується до тієї пори, поки увесь список букв не розіб'ється на частини, які містять по однієї букви.

Легко бачити, що при кожному подовженні коду, в одній частині коди подовжуються нулями, а в другій – одиницями. Отже, коди однієї частини не можуть бути префіксами іншої. Подовження коду закінчується тоді і тільки тоді, коли довжина частини дорівнює 1, тобто залишається єдиний код.

Таким чином, побудована схема кодування є префіксною, а значить розділимою.

Загальна схема побудови кодового дерева Фано.

Граф для коду Фано (кодове дерево) будується наступним чином. З верхньої (кореневої) вершини графа виходять два ребра, одне з яких позначене символом 0 , друге – символом 1. Ці два ребра відповідають розбиттю множині букв повідомлення на дві рівно імовірні підмножини. Кінцеві вершини цих ребер можна позначити тими сумарними ймовірностями, які відповідають розбиттю. Ребра, які виходять з вершин наступного ярусу теж відповідають розбиттю підмножин на рівно імовірні підмножини і т.д. Побудова графа закінчується, коли множина букв буде розбита на одноелементні підмножини. Кожна кінцева вершина графа, тобто вершина, з якої вже не виходять ребра, відповідає деякому кодовому слову. Щоб вказати це слово, треба пройти шлях

від кореневої вершини до відповідної кінцевої, виписуючи в порядку проходження по цьому шляху символи прохідних ребер.

12.2. Приклади виконання завдань.

Приклад 12.1. Для списку літер з заданим розподілом частот побудувати код Фано. Визначити вартість коду.

Літера	r	s	t	u	v	w	x	y	z
Ймовірність	0,10	0,05	0,12	0,04	0,03	0,11	0,32	0,08	0,15

Вибір медіани на кожному кроці можна простежити з таблиці, відокремлюючи коди з різним числом знаків.

Літера	*P*	Розбиття на підгрупи				Код
x	0,32	0	00			00
z	0,15		01			01
t	0,12	1	10	100		100
w	0,11			101		101
r	0,10		11	110	1100	1100
y	0,08				1101	1101
s	0,05			111	1110	1110
u	0,04				1111	11110
v	0,03					11111

Визначимо вартість коду:

$$l_\sigma(P) = 2 \cdot (0{,}32 + 0{,}15) + 3 \cdot (0{,}12 + 0{,}11) + 4 \cdot (0{,}10 + 0{,}08 + 0{,}05) + 5 \cdot (0{,}04 + 0{,}03) = 2{,}9.$$

Приклад 12.2. Закодувати по Фано:

Літера	1	2	3	4	5	6	7
Ймовірність	0,4	0,2	0,1	0,1	0,1	0,05	0,05

Побудову кодового дерева починаємо з кореня. На першому кроці літери поділено на підмножини з сумарними ймовірностями 0,4 та 0,6. Вершини першого ярусу позначено цими значеннями ймовірностей з вказанням відповідних літер. Далі, знову дивимось на вихідну таблицю та розбиваємо підмножину літер {2, 3, 4, 5, 6, 7} на підмножини {2, 3} та {4, 5, 6, 7}. Продовжуємо будувати дерево згідно правилу розбиття на більш рівно ймовірні підмножини літер.

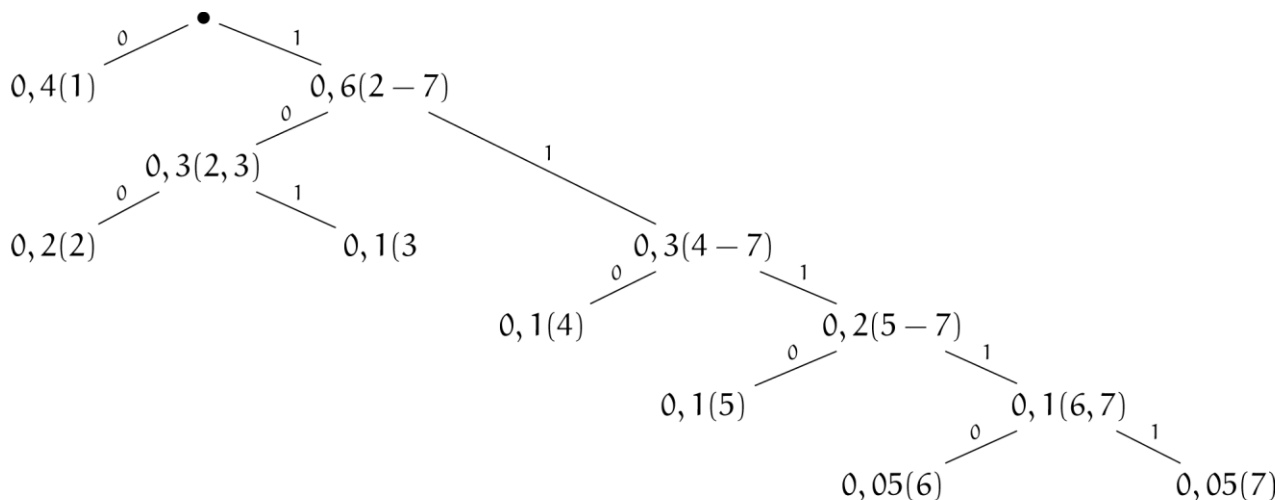


Рисунок 12.1 – Кодове дерево Фано

Отже кодування наступне:

Літера	1	2	3	4	5	6	7
Код	0	100	101	110	1110	11110	11111

12.3. Завдання для самостійного виконання.

1. Проведіть кодування по методу Фано алфавіту з чотирьох літер, ймовірності яких дорівнюють 0,4; 0,3; 0,2 й 0,1.

2. Алфавіт має 7 літер, які зустрічаються з ймовірностями 0,4; 0,3; 0,2; 0,1; 0,1; 0,05; 0,05. Виконайте кодування по методу Фано.

3. Алфавіт має дві літери, які зустрічаються з ймовірностями 0,8 й 0,2. Застосуйте код Фано до кодування всіх можливих дволітерних й три літерних комбінацій.

4. Проведіть кодування за методами Фано п'яти літер, які рівномірно зустрічаються.

5. Проведіть кодування двох літерних комбінацій чотирьох літер з завдання 1.

6. Порахуйте ефективність кода Фано при кодування алфавіту з 10 літер, які зустрічаються з ймовірностями 0,3; 0,2; 0,1; 0,1; 0,1; 0,05; 0,05; 0,04; 0,03; 0,03.

7. Порахуйте ефективність кода Фано при кодування алфавіту з 16 літер, які зустрічаються з ймовірностями 0,25; 0,2; 0,1; 0,1; 0,05; 0,04; 0,04; 0,04; 0,03; 0,03; 0,03; 0,03; 0,02; 0,02; 0,01; 0,01.

8. Для поданого розподілу ймовірностей побудувати двійкові коди, використовуючи метод Фано.

1) P = (0,2; 0,4; 0,2; 0,2);
2) P = (0,1; 0,1; 0,1; 0,7);
3) P = (0,2; 0,2; 0,2; 0,2; 0,2);
4) P = (0,08; 0,03; 0,09; 0,1; 0,5; 0,2);
5) P = (0,3; 0,4; 0,06; 0,08; 0,04; 0,04; 0,04; 0,04);
6) P = (0,3; 0,3; 0,03; 0,03; 0,03; 0,03; 0,03; 0,01; 0,2; 0,04);
7) P = (0,06; 0,06; 0,06; 0,06; 0,06; 0,3; 0,2; 0,1; 0,1);
8) P = (0,1; 0,2; 0,4; 0,05; 0,05; 0,05; 0,05; 0,05; 0,05);
9) P = (0,25; 0,15; 0,05; 0,1; 0,2; 0,1; 0,05);
10) P = (0,11; 0,08; 0,01; 015; 0,25; 0,21; 0,09; 0,1).

9. З'ясувати, чи є код V оптимальним для розподілу ймовірностей P = (0,15; 0,25; 0,05; 0,01; 0,09; 0,25; 0,15; 0,05).

1) V = (001; 010; 10; 11; 101; 011; 0110; 00);
2) V = (0000, 001; 1001; 10010; 0111; 111, 0010, 1110);
3) V = (000; 001; 010; 011; 100; 101; 110; 111);
4) V = (100; 00; 11110; 11111; 1110; 01; 101; 110);
5) V = (000; 01; 1110; 11111; 110; 10; 001; 11110);
6) V = (000; 10; 0111; 01111; 011; 01; 100; 11111).

ТЕМА 13.

ПОБУДОВА ОПТИМАЛЬНИХ КОДІВ МЕТОДОМ ХАФФМАНА.

Ключові слова: кодова таблиця Хаффмана, кодове дерево Хаффмана.

13.1. Основні теоретичні відомості.

Алгоритм побудови кодової таблиці Хаффмана.

1. Перший рядок таблиці *P* – це вихідні ймовірності, які записані в порядку зменшення.

2. Другий рядок *K* (основний) поки що залишають порожнім.

3. Прямий хід – заповнення рядків P_i зверху вниз, за кожним з яких залишають порожній рядок K_i.

Правило заповнення рядка P_i наступне: в рядку P_{i-1}, знаходять дві найменші ймовірності і їхня сума становиться новим елементом у рядку P_i та займає відповідне місце у списку спадних по величині імовірностей цього рядка.

2. Зворотний хід – заповнення рядків K_i знизу нагору.

В останньому *K*-рядку двом елементам, що залишилися, останнього *P*–рядка привласнюють коди 0 і 1.

Далі застосовують правило: два повідомлення з найменшими ймовірностями повинні мати однакові довжини кодових слів і розрізнятися останніми символами.

Вартість кода по Хаффману трохи менше ніж по Фано. Коди Фано по оптимальності уступають кодам Хаффмана, який показав, що його методика дає граничний стиск інформаційного тексту.

Загальна схема побудови кодового дерева Хаффмана.

Кодове дерево Хаффмана краще будувати спираючись на кодову таблицю, точніше на рядки P_i . При цьому, важливо бачити, які літери (або літера) відносяться до того або іншого значення ймовірності. З кожної вершини дерева (крім кінцевих) виходять рівно два ребра, які позначені як 0 та 1. При цьому символ 0 відповідає більшому сумарному значенню ймовірностей (або ймовірності). Як і при побудові дерева Фано, кожну вершину дерева Хаффмана краще позначити значенням ймовірності, яка відповідає певній підмножині літер. Кожна кінцева вершина відповідає одній з літер повідомлення. Щоб скласти код літери, треба пройти шлях від кореневої вершини до відповідної

кінцевої, виписуючи в порядку проходження по цьому шляху символи прохідних ребер.

13.2. Приклади виконання завдань.

Приклад 13.1. Для списку літер з заданим розподілом частот побудувати код Хаффмана. Визначити вартість коду.

Літера	r	s	t	u	v	w	x	y	z
Ймовірність	0,10	0,05	0,12	0,04	0,03	0,11	0,32	0,08	0,15

Побудуємо кодову таблицю Хаффмана:

	x	z	t	w	r	y	s	u	v
P	0,32	0,15	0,12	0,11	0,10	0,08	0,05	0,04	0,03
K	**00**	**010**	**011**	**101**	**110**	**111**	**1001**	**10000**	**10001**
P_1	0,32(x)	0,15(z)	0,12(t)	0,11(w)	0,10(r)	0,08(y)	0,07(u,v)	0,05(s)	
K_1	00	010	011	101	110	111	1000	1001	
P_2	0,32(x)	0,15(z)	0,12(t)	0,12(u,v,s)	0,11(w)	0,10(r)	0,08(y)		
K_2	00	010	011	100	101	110	111		
P_3	0,32(x)	0,18(r,y)	0,15(z)	0,12(t)	0,12(u,v,s)	0,11(w)			
K_3	00	11	010	011	100	101			
P_4	0,32(x)	0,23(u,v,s,w)	0,18(r,y)	0,15(z)	0,12(t)				
K_4	00	10	11	010	011				
P_5	0,32 (x)	0,27 (z,t)	0,23 (u,v,s,w)	0,18 (r,y)					
K_5	00	01	10	11					
P_6	0,41 (u,v,s,w,r,y)	0,32 (x)	0,27 (z,t)						
K_6	1	00	01						
P_7	0,59 (x,z,t)	0,41 (u,v,s,w,r,y)							
K_7	0	1							

Для зворотного ходу, з таблиці знаходимо, що ймовірність 0,59 (з кодом 0) рядка P_7 є сума ймовірностей 0,27 і 0,32 рядка P_6. Виходить, коди

ймовірностей 0,27 і 0,32 рядка P_6 будуть 00 і 01, відповідно. Продовжуємо заповнювати рядки P_i таким же чином.

Середня довжина слова, закодованого по Хаффману, дорівнює

$$l_\sigma(P) = 2 \cdot 0{,}32 + 3 \cdot (0{,}15 + 0{,}12 + 0{,}11 + 0{,}10 + 0{,}08) + 4 \cdot 0{,}05 + \\ + 5 \cdot (0{,}04 + 0{,}03) = 2{,}87$$

що трохи менше, ніж по Фано.

Приклад 13.2. Закодувати по Хаффману:

Літера	1	2	3	4	5	6	7
Ймовірність	0,4	0,2	0,1	0,1	0,1	0,05	0,05

Побудуємо кодову таблицю Хаффмана:

	1	2	3	4	5	6	7
P	0,4	0,2	0,1	0,1	0,1	0,05	0,05
K	**1**	**01**	**0010**	**0011**	**0000**	**00010**	**00011**
P_1	0,4(1)	0,2(2)	0,1(3)	0,1(4)	0,1(5)	0,1(6,7)	
K_1	1	01	0010	0011	0000	0001	
P_2	0,4(1)	0,2(2)	0,2(5,6,7)	0,1(3)	0,1(4)		
K_2	1	01	000	0010	0011		
P_3	0,4(1)	0,2(2)	0,2(5,6,7)	0,2(3,4)			
K_3	1	01	000	001			
P_4	0,4(1)	0,4(5,6,7,3,4)	0,2(2)				
K_4	1	00	01				
P_5	0,6(5,6,7,3,4,2)	0,4(1)					
K_5	0	1					

Побудова кодового дерева (рис. 13.1) починається з кореня зверху. Двом ребрам, що виходять з нього приписуються ваги 0 та 1. Згідно кодовій таблиці (рядок P_5) кінцеві вершини цих позначимо як 0,6(2-7) та 0,4(1).

Далі «піднімаємось» по кодовій таблиці на рядок P_4. Оскільки ймовірність 0,6 є результатом складання двох ймовірностей 0,4 і 0,2, з вершини 0,6(2-7) виходять два ребра з кінцевими вершинами 0,4(3-7) та 0,2(2). При цьому, вершина 0,2(2) буде кінцевою і 01 є код літери 2. Процедура продовжується до тих пір, поки в таблиці не досягнемо рядка P.

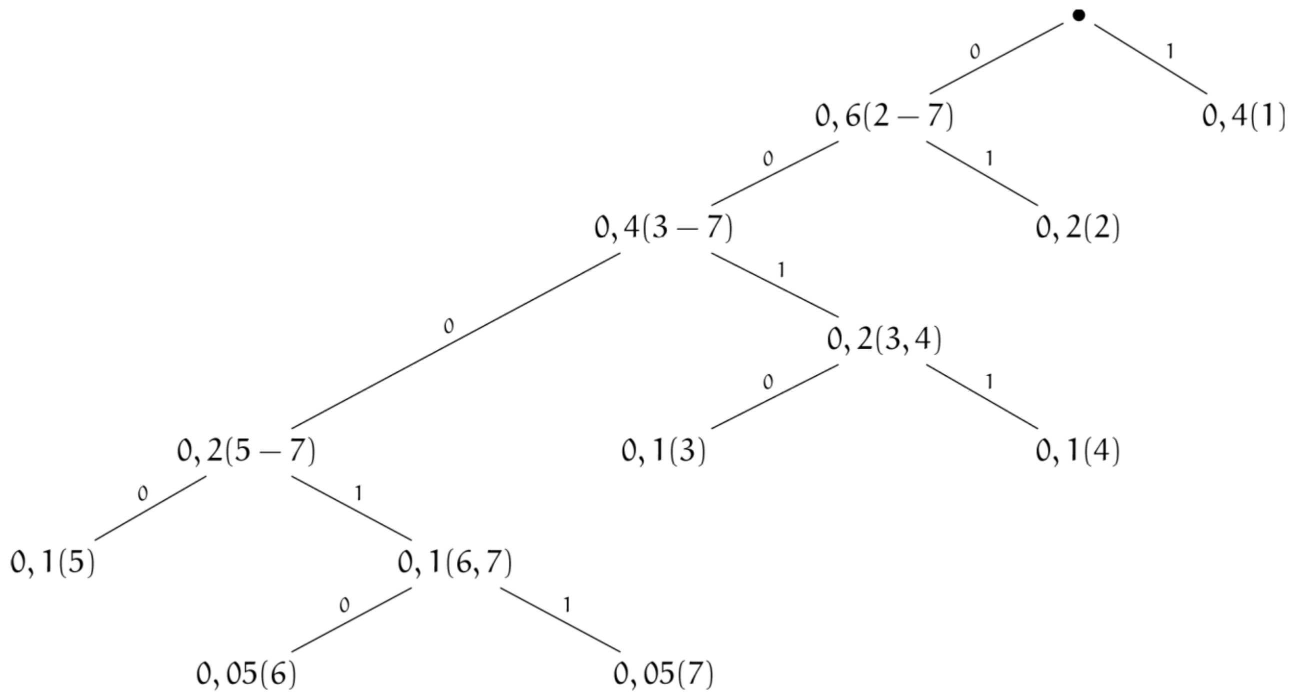


Рисунок 13.1 – Кодове дерево Хаффмана

Як бачимо, конфігурація кодового дерева Хаффмана для даного прикладу чітко відрізняється від кодового дерева Фано (рис. 12.1).

13.3. Завдання для самостійного виконання.

1. Проведіть кодування по методу Хаффмана алфавіту з чотирьох літер, ймовірності яких дорівнюють 0,35; 0,25; 0,23 й 0,17.

2. Алфавіт має 6 літер, які зустрічаються з ймовірностями 0,35; 0,25; 0,2; 0,1; 0,07; 0,03. Виконайте кодування по методу Хаффмана.

3. Алфавіт має дві літери, які зустрічаються з ймовірностями 0,7 й 0,3. Застосуйте код Хаффмана до кодування всіх можливих дволітерних й три літерних комбінацій.

4. Проведіть кодування за методами Хаффмана п’яти літер, які рівномірно зустрічаються.

5. Проведіть кодування двох літерних комбінацій чотирьох літер з завдання 1.

6. Порівняйте ефективність кодів Хаффмана при кодуванні алфавіту з 10 літер, які зустрічаються з ймовірностями 0,25; 0,2; 0,15; 0,13; 0,1; 0,07; 0,04; 0,03; 0,02; 0,01.

7. Порівняйте ефективність кодів Хаффмана при кодуванні алфавіту з 15 літер, які зустрічаються з ймовірностями 0,23; 0,22; 0,1; 0,09; 0,06; 0,05; 0,05; 0,04; 0,04; 0,03; 0,03; 0,02; 0,02; 0,01; 0,01.

8. Для поданого розподілу ймовірностей побудувати двійкові коди, використовуючи метод Хаффмана.

1) P = (0,3; 0,3; 0,2; 0,2);
2) P = (0,15; 0,05; 0,13; 0,67);
3) P = (0,25; 0,25; 0,15; 0,15; 0,2);
4) P = (0,07; 0,04; 0,08; 0,2; 0,5; 0,2);
5) P = (0,25; 0,35; 0,01; 0,06; 0,08; 0,04; 0,04; 0,05; 0,03);
6) P = (0,45; 0,25; 0,03; 0,03; 0,03; 0,03; 0,03; 0,01; 0,2; 0,04);
7) P = (0,06; 0,06; 0,06; 0,06; 0,06; 0,25; 0,25; 0,09; 0,11);
8) P = (0,1; 0,2; 0,3; 0,15; 0,05; 0,05; 0,05; 0,05; 0,05);
9) P = (0,15; 0,15; 0,05; 0,1; 0,25; 0,1; 0,2);
10) P = (0,12; 0,08; 0,01; 015; 0,25; 0,21; 0,09; 0,09).

9. З’ясувати, чи є код V оптимальним для розподілу ймовірностей P = (0,13; 0,27; 0,07; 0,01; 0,07; 0,25; 0,15; 0,05)

1) V = (001; 010; 10; 11; 101; 011; 0110; 00);
2) V = (0000, 001; 1001; 10010; 0111; 111, 0010, 1110);
3) V = (000; 001; 010; 011; 100; 101; 110; 111);
4) V = (100; 00; 11110; 11111; 1110; 01; 101; 110);
5) V = (000; 01; 1110; 11111; 110; 10; 001; 11110);
6) V = (000; 10; 0111; 01111; 011; 01; 100; 11111).

ТЕМА 14.

ЗАДАЧІ НА ВИКОРИСТАННЯ ОСНОВНИХ КОМБІНАТОРНИХ КОНФІГУРАЦІЙ ТА РЕКУРЕНТНИХ СПІВВІДНОШЕНЬ.

Ключові слова: сполучення, розміщення, перестановки, принцип включення та виключення, біном Ньютона, біноміальні коефіцієнти, рекурентні співвідношення, твірна функція (енумератор).

14.1. Основні теоретичні відомості.

Комбінаторна конфігурація – це підмножина дискретної множини елементів, яка задовольняє певним властивостям.

До основних комбінаторних конфігурацій відносяться сполучення, розміщення та перестановки. Інші конфігурації можуть бути зведені до основних.

Набір елементів $x_{i_1}, \ldots, x_{i_k}$, який складений з n елементів множини X, називається *вибіркою* обсягом k з n елементів, або (n, k)-вибіркою.

Упорядковану вибірку називають *словом*. Неупорядковані (n, k)-вибірки називають *сполученнями*. Якщо в (n, k)-вибірках елементи повторюються, то це *сполучення з повтореннями.* (n, k) – слова називають *розміщеннями*.

Якщо в (n, k) – слові елементи повторюються, то це (n, k) – *розміщення з повтореннями.*

(n, n) – розміщення називаються перестановками.

Для вибірки з повтореннями можливо $k > n$.

Формули перерахунку числа основних комбінаторних конфігурацій:

$C_n^k = \frac{n!}{k!(n-k)!}$ – число сполучень;

$H_n^k = C_{n+k-1}^k$ – число сполучень з повтореннями;

$A_n^k = \frac{n!}{(n-k)!}$ – число розміщень;

n^k – число розміщень з повтореннями;

$P_n = n!$ – число перестановок.

В загальному випадку, при фіксованому n та поточному k, послідовність чисел (n, k) називають *комбінаторними числами*.

Біном Ньютона.

Для будь-якого $n \in N$ і формальних змінних x та y має місце наступна рівність, яку називають біномом Ньютона:

$$(x+y)^n = \sum_{k=0}^{n} C_n^k x^k y^{n-k}.$$

У даному випадку, числа C_n^k називають *біномними коефіцієнтами*.

Деякі комбінаторні тотожності, які випливають з формули бінома Ньютона:

1. $C_n^0 + C_n^1 + \ldots + C_n^n = 2^n$.

2. $C_n^0 - C_n^1 + \ldots + (-1)^n C_n^n = 0$.

3. $\sum_{k=1}^{n} k \cdot C_n^k = n \cdot 2^{n-1}$.

4. $\sum_{k=2}^{n} k(k-1) C_n^k = n(n-1) 2^{n-2}$.

5. $(1+x)^n = \sum_{k=0}^{n} C_n^k x^n = \sum_{k=0}^{n} C_n^k x^{n-k}$.

6. $C_n^k = C_n^{n-k}$.

7. $C_n^i \cdot C_i^k = C_n^k \cdot C_{n-k}^{i-k}$.

8. $C_n^k = C_{n-1}^k + C_{n-1}^{k-1}$.

З останньою властивістю пов'язана схема, яка називається *трикутником Паскаля*:

n \ k	0	1	2	3	4	5	…
0	**1**						
1	**1**	**1**					
2	**1**	**2**	**1**				
3	**1**	**3**	**3**	**1**			
4	**1**	**4**	**6**	**4**	**1**		
5	**1**	**5**	**10**	**10**	**5**	**1**	
…							…

В цій схемі i-й рядок складається з чисел C_i^0, C_i^1, …, C_i^i. Кожний елемент рядка, крім крайніх, можна одержати як суму двох суміжних чисел, які знаходяться над взятим елементом у попередньому рядку. Це дає простий метод побудови трикутника Паскаля та знаходження біномних коефіцієнтів.

Принцип включення та виключення.

Нехай маємо N об'єктів та деяку сукупність властивостей $H = \{\alpha_1, \alpha_2, \ldots, \alpha_n\}$. Позначимо через $N(\alpha_i)$, $N(\alpha_i, \alpha_j)$, $N(\alpha_i, \alpha_j, \alpha_k)$ і так далі, кількість об'єктів, які мають відповідні властивості. Якщо треба підкреслити, що об'єкт не має властивості α_i, то пишуть $\overline{\alpha}_i$.

Кількість об'єктів, які не мають жодної властивості множини H, визначається *формулою включення та виключення*:

$$N(\bar{\alpha}_1\bar{\alpha}_2\ldots\bar{\alpha}_n) = N - \sum_i N(\alpha_i) + \sum_{i<j} N(\alpha_i\alpha_j) - \sum_{i<j<k} N(\alpha_i\alpha_i\alpha_k) + \ldots +$$
$$+ (-1)^n N(\alpha_1\alpha_2\ldots\alpha_n).$$

Якщо записати $\bar{\alpha}_i = 1 - \alpha_i$ та розглянути послідовність символів $\alpha_1\bar{\alpha}_2\ldots\bar{\alpha}_n$ як алгебраїчний добуток, то формулу включення та виключення можна представити в символічному виді.

Наприклад, для $n = 3$ маємо

$$N[(1-\alpha_1)(1-\alpha_2)(1-\alpha_3)] =$$
$$N[1-\alpha_1-\alpha_2-\alpha_3+\alpha_1\alpha_2+\alpha_1\alpha_3+\alpha_2\alpha_3-\alpha_1\alpha_2\alpha_3] =$$
$$= N - N(\alpha_1) - N(\alpha_2) - N(\alpha_3) +$$
$$+ N(\alpha_1\alpha_2) + N(\alpha_1\alpha_3) + N(\alpha_2\alpha_3) - N(\alpha_1\alpha_2\alpha_3).$$

Дякуючи такий формалізації можливо записати формулу для числа об’єктів, які одночасно мають деякі властивості, а деякі ні. Наприклад:

$$N[\alpha_1\bar{\alpha}_2\alpha_3] = N[\alpha_1(1-\alpha_2)\alpha_3] =$$
$$N[\alpha_1\alpha_3 - \alpha_1\alpha_2\alpha_3] = N(\alpha_1\alpha_2) - N(\alpha_1\alpha_2\alpha_3).$$

Рекурентні співвідношення.

Прикладом рекурентного співвідношення є наступна властивість біномних коефіцієнтів:

$$C_n^k = C_{n-1}^k + C_{n-1}^{k-1}.$$

Всяка рекурентність дає змогу знаходити наступні члени певної послідовності за рахунок попередніх. При цьому важливо вказати *граничні умови* для рекурентного співвідношення. Так, для приведеної рекурентності, граничними умовами є:

$$C_n^0 = C_1^1 = 1, \quad C_n^s = 0 \quad \forall n < s .$$

Розв'язати рекурентне співвідношення – значить одержати вираз для k-го члена послідовності у явному виді. Наприклад, для послідовності біномних коефіцієнтів $C_n^o, \ldots, C_n^k, \ldots, C_n^n$ маємо $C_n^k = \dfrac{n!}{k!(n-k)!}$.

Твірні функції.

З усякою послідовністю *комбінаторних чисел* a_0, a_1, a_2, можна зв'язати суму (ряд) виду:

$$f(x) = \sum_{k=0}^{\infty} a_k x^k,$$

яка називається *твірною функцією* або *енумератором* для даної послідовності.

Біном Ньютона у вигляді $(1+x)^n = \sum_{k=0}^{n} C_n^k x^k$ є енумератором для послідовності $C_n^0, C_1^1, \ldots, C_n^n$, тобто для числа сполучень з n елементів.

Якщо записати біном Ньютона наступним чином:

$$(1+x)^n = \sum_{k=0}^{n} C_n^k x^k = \sum_{k=0}^{n} A_n^k \frac{x^k}{k!},$$

то отримаємо енумератор для числа розміщень з n елементів, який є *експонентною твірною функцією*.

14.2. Приклади виконання завдань.

Приклад 14.1. Задача про ханойську вежу.

Нехай маємо два вільних кілочки і вежу з *n* дисків, нанизаних на кілочок, таких що розмір дисків зверху униз зростає, тобто нижній диск – найбільший.

Елементарним кроком вважаємо перенесення одного диска з одного кілочку на другий.

Задача полягає в тому, щоб за мінімальну кількість елементарних кроків перемістити на один з вільних кілочок усю вежу, зберігаючи порядок дисків по зростанню їх розміру зверху униз. При цьому, при перенесенні дисків більший диск не можна класти на менший.

Якщо T_n – шукана мінімальна кількість кроків, то

$$\begin{cases} T_0 = 0; \\ T_n = 2T_{n-1} + 1. \end{cases}$$

Дійсно, спочатку перекладаємо за T_{n-1} кроків n–1 дисків на будь-який вільний кілочок, при цьому буде збережемо порядок відносно розмірів дисків. Далі, перекладаємо останній найбільший диск на другий вільний кілочок і знову перекладаємо n–1 дисків. Отже, маємо рекурентність $T_n = 2T_{n-1} + 1$.

Обчислимо декілька перших значень T_n:

$T_0 = 0,\ T_1 = 1,\ T_2 = 3,\ T_3 = 7,\ T_4 = 15, T_5 = 31.$

Припустимо, що $T_n = 2^n - 11$ и застосуємо метод математичної індукції. Нехай $T_{n-1} = 2^{n-1} - 1$, тоді:

$$T_n = 2T_{n-1} + 1 = 2\left(2^{n-1} - 1\right) + 1 = 2^n - 1.$$

Отже, маємо розв'язок даної рекурентності: $T_n = 2^n - 1$.

Приклад 14.2. Числа Фібоначі.

Числа Фібоначі визначаються наступним рекурентним співвідношенням:

$$F_n = F_{n-1} + F_{n-2},\ n \geq 2\ ,$$

з граничними умовами: $F_0 = F_1 = 1$.

Вираз $F_n = F_{n\text{-}1} + F_{n\text{-}2}$ можна переписати так:

$$F_{n+2} - F_{n+1} - F_n = 0\,.$$

Це приклад лінійної однорідної рекурентності. Коефіцієнти цього лінійного однорідного рівняння є: $a_0 = 1$, $a_1 = -1$, $a_2 = -1$.

Однорідне лінійне рекурентне рівняння з постійними коефіцієнтами у загальному випадку має вид:

$$F(n+k) + a_1F(n+k-1) + a_2F(n+k-2) + \ldots + a_n F(n) = 0\,.$$

Відповідно, характеристичне рівняння має вид:

$$\lambda^k + a_1\lambda^{k-1} + a_2\lambda^{k-2} + \ldots + a_n = 0.$$

Тоді, функція $F(n) = C_1\lambda_1^n + C_2\lambda_2^n + \ldots + C_k\lambda_k^n$ є загальним розв'язком рекурентного рівняння, де $C_1, C_2, \ldots, C_k$ – довільні постійні, $\square_1^n, \square_2^n, \ldots_k, \square^n$ – корені характеристичного рівняння (можливо, комплексні). Використовуючи граничні умови, можна одержати частинний розв'язок.

Неоднорідне лінійне рекурентне рівняння має вид:

$$F(n+k) + a_1F(n+k-1) + a_2F(n+k-2) + \ldots + a_n F(n) = b\,.$$

Його загальний розв'язок є сума частинного розв'язку і загального розв'язку відповідного йому однорідного рівняння.

Частинний розв'язок у випадку $1 + \sum_{i=1}^{k} a_i \neq 0$ знаходять з рівняння:

$$\xi + a_i\xi + \ldots + a_k\xi = b\,;\; \xi = \frac{b}{1 + \sum_{i=1}^{k} a_i}\,.$$

Тоді,

$$F(n) = C_1\lambda_1^n + \ldots + C_k\lambda_k^n + \frac{b}{1 + \sum_{i=1}^{k} a_i}\,.$$

Використаємо твірну функцію для знаходження в явному виді чисел Фібоначі.

Нехай $f_0, f_1, \ldots, f_n, \ldots$ – послідовність чисел Фібоначі, така що $f_0 = f_1 = 1$; $f_s = f_{s-1} + f_{s-2}$.

Тоді,

$$F(x) = \sum_{k=0}^{\infty} f_k x^k = 1 + x + \sum_{k=2}^{\infty} (f_{k-2} + f_{k-1}) x^k =$$

$$1 + x + x^2 \sum_{k=2}^{\infty} f_{k-2} x^{k-2} + x \sum_{k=2}^{\infty} f_{k-1} x^{k-1} =$$

$$1 + x + x^2 F(x) + x(F(x) - 1) = 1 + \left(x + x^2\right) F(x).$$

Маємо рівняння:

$$F(x) = 1 + (x^2 + x)F(x),$$

з якого виходить, що

$$F(x) = \left(1 - x - x^2\right)^{-1}.$$

Приклад 14.3. Розв'язати рекурентне співвідношення, яким визначаються числа Фібоначі.

Для розв'язку рекурентного співвідношення $F_{n+2} - F_{n+1} - F_n = 0$ застосуємо метод характеристичного рівняння.

Характеристичним рівнянням даного рекурентного співвідношення є

$$\lambda^2 - \lambda - 1 = 0.$$

Характеристичні корні: $\lambda_{1,2} = \dfrac{1 \pm \sqrt{5}}{2}$.

Загальний розв'язок рекурентного рівняння має вид: $F_n = C_1 \lambda_1^n + C_2 \lambda_2^n$.

Використовуючи умови $F_0 = F_1 = 1$, одержимо систему:

$$\begin{cases} C_1 + C_2 = 1; \\ C_1 \lambda_1 + C_2 \lambda_2 = 1. \end{cases}$$

Тоді,

$$C_1 = \frac{1-\lambda_2}{\lambda_1 - \lambda_2} = \frac{1}{\sqrt{5}}\frac{1+\sqrt{5}}{2};$$

$$C_2 = 1 - C_1 = -\frac{1}{\sqrt{5}}\frac{1-\sqrt{5}}{2}.$$

Остаточно, маємо

$$F_n = \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^{n+1} - \frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^{n+1}.$$

Приклад 14.4. Знайти n, якщо $\frac{P_{n+5}}{P_{n-k}} = 240 \cdot A_{n+3}^{k+3}$, $k \le n$.

Застосовуючи формулу для числа перестановок та числа розміщень, запишемо рівняння наступним чином:

$$\frac{(n+5)!}{(n-k)!} = 240 \cdot \frac{\square(n+3)!}{(n+3-k-3)!}.$$

Отримане рівняння зводимо до квадратного $(n+5)(n+4) = 240$, корені якого $n = 11$ та $n = -20$. При $n = -20$ обидві частини рівняння не мають сенсу, тому при $n = 11$ для будь-якого k такого, що $0 \le k \le 11$, вірна рівність

$$\frac{P_{16}}{P_{11-k}} = 240 \cdot A_{14}^{k+3}.$$

Тобто, $n = 11$.

Приклад 14.5. Вирішити нерівність $C_{10}^{x-1} > 2C_{10}^{x}$.

Ліва частина нерівності має сенс тоді й тільки тоді, коли $x-$ціле число, яке належить відрізку $[1,11]$. Права частина має сенс тільки у тому випадку, коли $x-$ ціле число та $x \in [0,10]$.Тобто, рішення нерівності є цілі значення x, які належать відрізку $[1, 10]$.

Використовуючи формулу числа сполучень нерівність має вид:

$$\frac{10!}{(x-1)!(10-x+1)!} > 2\frac{10!}{x!(10-x)!}.$$

Виконавши необхідні перетворення, отримуємо $\frac{1}{11-x} > \frac{2}{x}$. Звідки $x > \frac{22}{3}$. Враховуючи область допустимих значень $x \in N$ та $x \in [1,10]$, отримуємо множину рішень нерівності: $\{8,9,10\}$.

Приклад 14.6. Скільки різних слів можна зіставити, якщо переставляти літери слова «математика»?

Всього в слові «математика» 10 літер, причому літера «м» зустрічається 2 рази, літера «а» – 3 рази, літера «т» – 2 рази, літера «е» – 1 раз, літера «и» – 1 раз, літера «к» – 1 раз. Тоді число всіх різних слів, котрі можна зіставити, дорівнює

$$P_{10}(2, 3, 2, 1, 1, 1) = \frac{10!}{2!3!2!} = 151200.$$

Приклад 14.7. З цифр 1, 2, 3, 4, 5 зіставляють будь-які числа, кожне з яких має не менш трьох цифр. Скільки таких чисел можна зіставити, якщо повторення цифр в числах заборонені?

Необхідно порахувати скільки існує трьох –, чотирьох –, п'яти значущих чисел, які зіставлені з цих п'яти цифр. Трьох значущих чисел $A_5^3 = \frac{5!}{(5-3)!} = 3 \cdot 4 \cdot 5 = 60$, чотирьох значущих чисел $A_5^4 = 120$ та п'яти значущих чисел $A_5^5 = 120$. Використовуючи принцип додавання маємо:

$$A_5^3 + A_5^4 + A_5^5 = 60 + 120 + 120 = 300 .$$

Приклад 14.8. Скількома способами з восьми чоловік можна обрати комісію з п'яти членів?

Використовуємо формулу сполучень елементів, так як не має значення порядок елементів в виборці. Тому

$$C_8^5 = \frac{8!}{5!(8-5)!} = \frac{6 \cdot 7 \cdot 8}{2 \cdot 3} = 56 .$$

Приклад 14.9. Нехай задані властивості: α_1 – стальний; α_2 – чорний; α_3 – сферичний та $N(\alpha_1) = 13$; $N(\alpha_2) = 10$; $N(\alpha_3) = 14$; $N(\alpha_1\alpha_2) = 4$; $N(\alpha_1\alpha_3) = 5$; $N(\alpha_2\alpha 3) = 3$; $N(\alpha_1\alpha_2\alpha_3) = 1$.

Якщо $N = 38$, то кількість стальних, але не чорних та не сферичних предметів, дорівнює

$$N(\alpha_1\bar{\alpha}_2\bar{\alpha}_3) = N[\alpha_1(1-\alpha_2)(1-\alpha_3)] =$$
$$= N(\alpha_1) - N(\alpha_1\alpha_2) - N(\alpha_1\alpha_3) + N(\alpha_1\alpha_2\alpha_3) = 13–4–5+1 = 5.$$

Приклад 14.10. Знайти кількість тризначних чисел, сума цифр яких дорівнює 20.

Позначимо через x, x_2, x_3 відповідно першу, другу та третю цифри в довільному тризначному числі $a = x_1 \cdot 100 + x_2 \cdot 10 + x_3$. Рішення задачі зводиться до знаходження цілих наборів (x_1, x_2, x_3), для яких

$$x_1 + x_2 + x_3 = 20,$$
$$1 \le x_1 \le 9,\ 0 \le x_2 \le 9,\ 0 \le x_3 \le 9.$$

Позначимо через X множину наборів (x_1, x_2, x_3), у яких

$$x_1 + x_2 + x_3 = 20,\ x_1 \ge 1,\ x_2 \ge 0,\ x_3 \ge 0,$$

та для елементів заданої множини введемо три властивості: $\alpha_1 - \ x_1 \ge 10$, $\square_2 - x_2 \ge 10$, $\alpha_3 - \ x_3 \ge 10$. Таким чином, в задачі необхідно знайти множину наборів, для яких не виконується жодна з цих властивостей.

Спочатку знайдемо загальну кількість елементів N в множині X . Якщо ввести змінні $y_1 = x_1 - 1, y_2 = x_2 , y_3 = x_3$, то число елементів в множині X можна знайти як число цілих наборів (y_1, y_2 , y_3) таких, що

$$y_1 + y_2 + y_3 = 19, \quad y_1 \geq 0 , \ y_2 \geq 0, \quad y_3 \geq 0.$$

Кількість таких наборів знаходимо за формуло для числа сполучень з повтореннями з 19 по 3, тобто

$$N = H_3^{19} = C_{19+3-1}^{19} = \frac{21!}{19!2!} = 10 \cdot 21 = 210.$$

Кількість $N(\alpha_1)$ співпадає з кількістю наборів (x_1, x_2, x_3), дляяких

$$x_1 + x_2 + x_3 = 20, \ x_1 \geq 10, \ x_2 \geq 0 , \ x_3 \geq 0,$$

кількість $N(\alpha_2)$ – з кількістю наборів (x_1, x_2,x_3), для яких

$$x_1 + x_2 + x_3 = 20, \ x_1 \geq 1, \ x_2 \geq 10, \ x_3 \geq 0,$$

кількість $N(\alpha_3)$ – з кількістю наборів (x_1, x_2,x_3), для яких

$$x_1 + x_2 + x_3 = 20, \ x_1 \geq 1, \ x_2 \geq 0 , \ x_3 \geq 10,$$

кількість $N(\alpha_1,\alpha_2)$ – з кількістю наборів (x_1, x_2,x_3), для яких

$$x_1 + x_2 + x_3 = 20, \ x_1 \geq 10, \ x_2 \geq 10, \ x_3 \geq 0,$$

кількість $N(\alpha_1,\alpha_3)$– з кількістю наборів (x_1, x_2, x_3), для яких

$$x_1 + x_2 + x_3 = 20, \ x_1 \geq 10, \ x_2 \geq 0 , \ x_3 \geq 10,$$

кількість $N(\alpha_2 ,\alpha_3)$ – з кількістю наборів (x_1, x_2, x_3), для яких

$$x_1 + x_2 + x_3 = 20, \ x_1 \geq 1, \ x_2 \geq 10, \ x_3 \geq 10,$$

кількість $N(\square_1,\square_2,\square_3)$ – з кількістю наборів (x_1, x_2, x_3), для яких

$$x_1 + x_2 + x_3 = 20,\ \ x_1 \geq 10,\ \ x_2 \geq 10,\ \ x_3 \geq 10.$$

Використовуючи необхідні заміни змінних та формулу сполучень з повтореннями, знайдемо

$$N(\alpha_1) = N_3^{20-10} = C_{10+3-1}^{10} = C_{12}^{10} = \frac{12!}{10!2!} = 66,$$

$$N(\alpha_2) = N_3^{20-11} = C_{9+3-1}^{9} = C_{11}^{9} = \frac{11!}{9!2!} = 55\ ,$$

$$N(\alpha_3) = N_3^{20-11} = C_{9+3-1}^{9} = C_{11}^{9} = \frac{11!}{9!2!} = 55,$$

$$N(\square_1,\square_2) = N_3^{20-20} = 1,$$

$$N(\square_1,\square_3) = N_3^{20-20} = 1,$$

$$N(\square_2,\square_3) = N_3^{20-21} = 0,$$

$$N(\square_1,\square_2,\square_3) = 0.$$

Тоді,

$$N_0 = N - N(\alpha_1) - N(\alpha_2) - N(\alpha_3) + N(\alpha_1,\alpha_2) + N(\alpha_1,\alpha_3) +$$
$$+ N(\square_2,\square_3) - N(\square_1,\square_2,\square_3) = 210 - 66 - 55 - 55 + 1 + 1 = 36.$$

Приклад 14.11. Обчислити $\left(1 + x + x^2\right)^3$.

За поліноміальною формулою маємо:

$$\left(1 + x + x^2\right)^3 = \sum \frac{3!}{k_1!k_2!k_3!} 1^{k_1} x^{k_2} \left(x^2\right)^{k_3},$$

де додавання ведеться по всім наборам невід'ємних цілих чисел k_1, k_2, k_3, для яких $k_1 + k_2 + k_3 = 3$.

Випишемо такі набори: (0, 0, 3), (0, 3, 0), (3, 0, 0), (0, 1, 2), (1, 2, 0), (2, 0, 1), (1, 0, 2), (0, 2, 1), (2, 1, 0), (1, 1, 1).

Тепер знаходимо:

$$\left(1 + x + x^2\right)^3 = \sum \frac{3!}{k_1!k_2!k_3!} 1^{k_1} x^{k_2} \left(x^2\right)^{k_3} =$$

$$= \frac{3!}{0!0!3!}\left(x^2\right)^3 + \frac{3!}{0!3!0!} x^3\left(x^2\right)^0 + \frac{3!}{3!0!0!} x^0\left(x^2\right)^0 + \frac{3!}{0!1!2!} x^1\left(x^2\right)^2 + \frac{3!}{1!2!0!} x^2\left(x^2\right)^0 +$$

$$+ \frac{3!}{2!0!1!} x^0\left(x^2\right)^1 + \frac{3!}{1!0!2!} x^0\left(x^2\right)^2 + \frac{3!}{0!2!1!} x^2\left(x^2\right)^1 + \frac{3!}{2!1!0!} x^1\left(x^2\right)^0 + \frac{3!}{1!1!1!} x^1\left(x^2\right)^1 =$$

$$= x^6 + 3x^5 + 6x^4 + 7x^3 + 6x^2 + 3x + 1.$$

Приклад 14.12. Знайти в розкладі бінома $\left(x + \frac{2}{x}\right)^{12}$ член з нульовим степенем x.

$$C_n^k x^{n-k} y^k = C_{12}^6 x^{12-6} \left(\frac{2}{x}\right)^6 = C_{12}^6 x^6 \frac{64}{x^6} = \frac{12!}{6!6!} \cdot 64 = 59136.$$

14.3. Завдання для самостійного виконання.

1. Скількома способами 8 чоловік можуть утворити чергу?

2. В автомобілі 7 місць. Скількома способами 7 чоловік можуть зайняти в ній місця, якщо місця водія можуть зайняти тільки троє?

3. Алфавіт деякої мови має 30 літер. Скільки існує шести літерних слів, які отримані за допомогою літер цього алфавіту, якщо:

а) літери в словах не повторюються;

б) літери в словах повторюються.

4. Скільки існує різних автомобільних номерів, які мають 5 цифр, якщо:

а) перша з них не дорівнює нулю;

б) якщо номер має одну літеру латинського алфавіту, за якою йдуть чотири цифри, які не дорівнюють нулю.

5. Компанія з 20 чоловіків об'єднується в групи. В першу входить 3 чоловіка, в другу – 5, в третю – 12. Скількома способами вони можуть це зробити?

6. Скільки п'яти літерних слів, кожне з яких має три приголосні та дві голосні літери, можна утворити зі слова «математика».

7. Скільки різних перестановок можна утворити зі всіх літер слова «перестановка»? Скільки з них починається з літери «п» й закінчується літерою «а»?

8. Об'єднання множин A та B має 25 елементів, а їх перетин – 10 елементів. Скільки елементів с множині B, якщо множина A має 20 елементів?

9. Скільки існує додатних цілих чисел менших 1000, які

а) діляться і на 3, і на 7;

б) діляться на 3, але не діляться на 7;

в) діляться на 7, але не діляться на 3;

г) діляться на 3 або на 7;

д) не діляться ні на 3, ні на 7.

10. В хімічній олімпіаді приймали участь 21 чоловік, в фізичній – 26 чоловік, в математичній – 29 чоловік. 14 студентів приймали участь і в хімічній, і в математичній олімпіаді, 15 студентів – і в фізичній, і в математичній, 8 – у всіх трьох. Скільки студентів приймали участь хоча б одній олімпіаді? Знайдіть всі можливі відповіді.

11. На полу площиною $20\,м^2$ лежать три ковдри. Площина однієї ковдри $10\,м^2$, другої – $8\,м^2$, третьої – $5\,м^2$. Кожні дві ковдри перекриваються на площині $3\,м^2$. Всі три ковдри перекриваються на площині $1\,м^2$.

а) яка площина полу, яка не закрита ковдрами?

б) яка площина полу, яка закрита тільки однією ковдрою?

12. Доведіть тотожність:

а) $\sum_{k=0}^{n} C_n^k = 2^n, \quad n \ge 0$;

б) $\sum_{k=0}^{n} kC_n^k = n2^{n-1}$;

в) $\sum\limits_{k=0}^{n} k^2 C_n^k = n(n+1)2^{n-2}$;

г) $\sum\limits_{x=0}^{n} \frac{C_n^x C_n^r}{C_{2n}^{x+r}} = \frac{2n+1}{n+1}$;

д) $\frac{\left(C_{n+1}^{r+1} - C_n^r\right) C_{n-1}^{r-1}}{\left(C_n^r\right)^2 - C_{n+1}^{r+1} C_{n-1}^{r-1}} = r$, $n > 1, 0 < r < n$;

е) $\sum\limits_{x=1}^{n} \frac{C_{n-1}^{x-1}}{C_{2n-1}^{x-1}} = \frac{2}{n+1}$, $n \geq 1$.

13. Знайти n, якщо

а) $C_{n+4}^{n+1} - C_{n+3}^{n} = 15(n+2)$;

б) $\frac{1}{C_4^n} = \frac{1}{C_5^n} + \frac{1}{C_6^n}$;

в) $5C_n^3 = C_{n+2}^4$;

г) $(n+2)! = 132 A_n^k P_{n-k}$;

д) $C_{n+3}^{n+1} - 5C_{3n}^2 + 19n^2 = 6$;

е) $\frac{A_{n+4}^4}{(n+2)!} < \frac{143}{4P_n}$.;

14. Знайдіть в розкладі бінома $\left(a + \frac{1}{a}\right)^{12}$ коефіцієнт при a^8.

15. Знайдіть четвертий член в розкладі бінома $(8x - 5y)^6$.

16. Знайдіть ступень двочлена $\left(x^2 - \frac{1}{4}\right)^n$, якщо коефіцієнт при x^2 дорівнює 31.

17. В розкладі бінома $\left(x^2 + \frac{1}{x}\right)^m$ сума коефіцієнтів першого, другого та третього членів дорівнює 46. Знайти член розкладу, який не містить x.

18. Додаток біноміальних коефіцієнтів в розкладі біному $\left(\frac{2x}{3} + \frac{3}{2x^2}\right)^n$ дорівнює 64. Знайдіть член розкладу, який не містить x.

19. Використовуючи характеристичне рівняння, вирішити наступну рекурентність: $T_0 = 0$, $T_n = 2T_{n-1} + 1$ $(n > 0)$.

20. Використовуючи характеристичне рівняння, вирішити наступну рекурентність: $F(n+1) = F(n) + F(n-1)$, при $F(0) = 1, F(1) = 2$.

ТЕМА 15.

ОСНОВНІ ОЗНАЧЕННЯ ТЕОРІЇ ГРАФІВ. МАТРИЦІ ГРАФІВ.

Ключові слова: граф, носій графа, сигнатура графу, порядок графа, петля, кратні ребра, частковий граф, підграф, неорієнтований (орієнтований) граф, повний граф, степінь вершини, степінь графа, регулярний граф, простий ланцюг, відстань між вершинами, матриця суміжності, контур, матриця інциденцій, ваги дуг.

15.1. Основні теоретичні відомості.

Граф G – це пара $\langle V, \Gamma \rangle$, де V – множина вершин (носій графу), $\Gamma \subset V \times V$ – множина ребер (сигнатура графу), причому граф може містити ізольовані вершини. Кількість вершин графу $n(G)=|V|$ (потужність носія) називають *порядком графу.*

Ребро виду (v,v) називають *петлею* (ребро, яке з'єднує вершину саму з собою).

Граф з кратними ребрами без петель називають *мультиграфом*. Граф з петлями та кратними ребрами називають *псевдографом*.

Нехай $(v_i, v_j) \in \Gamma$, де v_i, $v_j \in V$, то, кажуть, що ребро (v_i, v_j) *інцидентно* вершинам v_i та v_j, а самі вершини *коінцидентні* ребру.

Якщо з графу вилучити деякі ребра, то одержимо *частковий граф*. При вилученні з графу вершин та інцидентних їм ребер одержимо *підграф* графу.

Два *ребра* називають *суміжними*, якщо вони інцидентні одній вершині.

Дві *вершини* називають *суміжними*, якщо вони інцидентні одному ребру (тобто, з'єднані ребром).

Граф називається *повним*, якщо кожна пара його вершин суміжна (рис. 15.1).

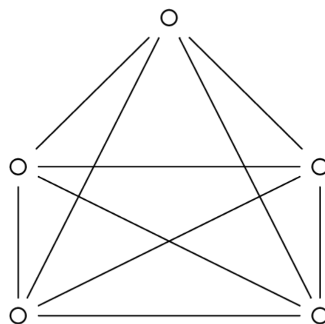


Рисунок 15.1 – Повний граф P_5 порядку 5

Кількість ребер графу, інцидентних деякій вершині v_α, називається *степенем* вершини і позначається $s(v_\alpha)$.

Максимальний степінь вершин графа G називається *степенем графа* і позначається $s(G)$.

Твердження 1.1. Сума степеней вершин будь-якого графу дорівнює подвійному числу його ребер.

Твердження 1.2. Кількість вершин непарного степеня будь-якого графу завжди парна.

Твердження 1.3. В будь-якому графі завжди знайдеться дві або більше вершин з однаковим степенем.

Якщо всі вершини графа мають однаковий степінь n, то такий граф називають *регулярним графом* степеня n. Відомий граф Петерсена (рис. 15.2) є регулярним графом степеня 3.

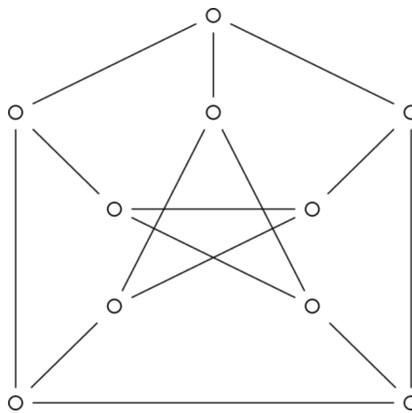


Рисунок 15.2 – Граф Петерсена

Послідовність ребер $(\gamma_1, \gamma_2, \ldots, \gamma_n)$ виду $\gamma_i = (v_i, v_{i+1})$ називають *маршрутом*.

Маршрут називається *ланцюгом*, якщо кожне ребро зустрічається в ньому не більше ніж один раз.

Простим ланцюгом називають такий ланцюг, в якому будь-яка вершина зустрічається в ньому не більш як один раз.

Замкнений ланцюг називають *простим циклом*.

Кількість ребер ланцюга, з'єднуючих вершини v_i та v_j називається його *довжиною* і позначається $l(v_i, v_j)$.

Мінімальна довжина ланцюга, з'єднуючого вершини v_i та v_j називається *відстанню* між цими вершинами і позначається $r(v_i, v_j)$.

Якщо ті пари вершин, які поєднані ребрами, тобто $(v, w) \in \Gamma$, вважаються упорядкованими, то граф називають *орієнтованим*. Ребра орієнтованого графу (орграфу) називають *дугами*, а напрям ребер позначають стрілками. При цьому, вершину v називають початком дуги (v, w), а вершину w – кінцем.

Ланцюг в орграфі називають *шляхом*. Будь-яка внутрішня вершина шляху є одночасно кінцем попередньої та початком наступної дуги.

Простий цикл в орграфі називають *контуром*.

Граф називається *зв'язним*, якщо будь-яка пара його вершин поєднана ланцюгом.

Зв'язаністю графу G називається найменше число вершин $\psi(G)$, вилучення яких робить граф незв'язним.

Реберною зв'язаністю графу G називається найменше число ребер $\lambda(G)$, вилучення яких робить граф незв'язним.

Дводольним графом $G(V_1, V_2)$ називається граф, множина вершин якого розбивається на дві не перетинаючи підмножини V_1, V_2 так, що кожне ребро графу з'єднує дві вершини з різних підмножин. Повний дводольний граф називається *зірковим* і позначається $K_{m,n}$, де $m = |V_1|$, $n = |V_2|$. Повний дводольний граф зображено на рис. 15.3. Граф $K_{m,n}$ має $(m + n)$ вершин та $m \cdot n$ ребер.

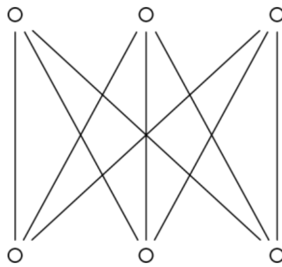


Рисунок 15.3 – Повний дводольний граф $K_{3,3}$ порядку 6

Твердження 1.4. Граф є дводольним тоді і тільки тоді, коли всі його цикли мають парну довжину.

Множина ребер графу, в якої ніяка пара не суміжна, називається *паросполученням* графу.

Гамільтоновим циклом (ланцюгом) називається простий цикл (ланцюг), що проходить через усі вершини графа. Граф, який містить гамільтоновий цикл (ланцюг), називається гамільтоновим. Прикладом гамільтонового графу є повний граф P_5, зображений на рис. 15.1. Граф, який містить гамільтонів цикл зображено на рис. 15.4. Граф, який містить гамільтонів ланцюг зображено на рис. 15.5.

Твердження 1.5. У повному графі порядку n існує рівно $(n-1)!$ гамільтонових циклів.

Твердження 1.6. Майже усі графи є гамільтоновими.

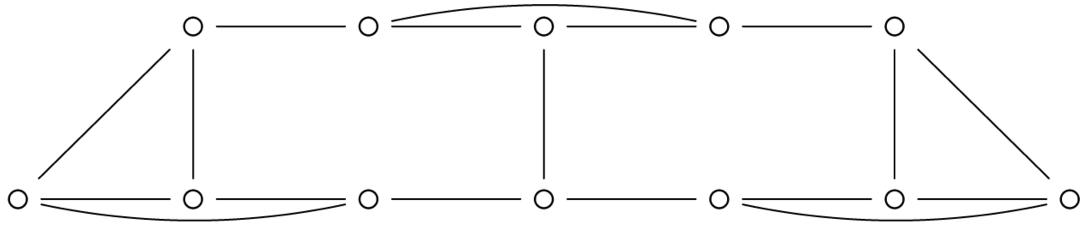


Рисунок 15.4

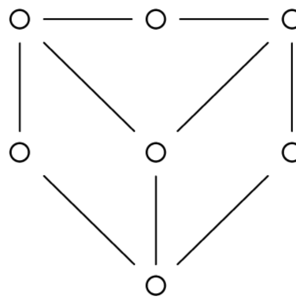


Рисунок 15.5

Два графа одного порядку $G = \langle V, \Gamma \rangle$ та $G' = \langle V', \Gamma' \rangle$ називаються *ізоморфними*, якщо існує така бієкція *f*, що для будь-яких вершин $v_1, v_2 \in V$ їхні образи $f(v_1),\ f(v_2) \in V'$ є суміжними тоді і тільки тоді, коли суміжні їх прообрази. Приклад ізоморфних графів приведено на рис. 15.6.

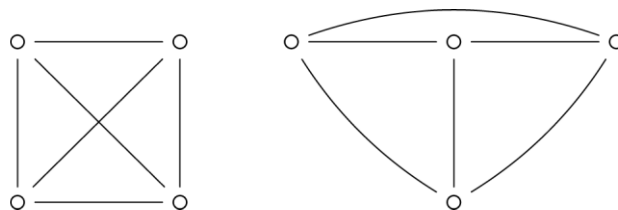


Рисунок 15.6

Граф *G* називається *плоским*, якщо його можна зобразити на площині без перетину ребер.

Очевидно, повний граф порядку 4 є плоским (рис. 15.6).

Граф G називається *планарним*, якщо він ізоморфний деякому плоскому графу.

Два графа називаються *гомеоморфними*, якщо після вилучення в них вершин другого степеня з подальшим об’єднанням інцидентних цим вершинам ребер вони стають ізоморфними.

Очевидно граф, гомеоморфний планарному графу, також є планарним, і навпаки, граф, гомеоморфний непланарному графу, – непланарний.

Твердження 1.7 (Критерій Понтрягіна-Куратовського). Граф G є планарним тоді і тільки тоді, коли він не містить підграфів, гомеоморфних одному з графів Понтрягіна-Куратовського (рис. 15.1, рис. 15.3).

Матрицею інцидентності неорієнтованого графу G порядку n та потужністю сигнатури m називається така прямокутна матриця $A(G) = \left[a_{ij}\right]_{m \times n}$, що

$$a_{ij} = \begin{cases} 1, & \text{якщо ребро } \square_i \text{ інцидентно вершині } v_j; \\ 0, & \text{в іншому випадку}. \end{cases}$$

Матрицею суміжності графу (або орграфу) G, який має порядок n, називається така квадратна матриця $S(G) = \left[s_{ij}\right]_{n \times n}$, що

$$s_{ij} = \begin{cases} 1, & \text{якщо} \left(v_i, v_j\right) \in \Gamma, \\ 0, & \text{якщо} \left(v_i, v_j\right) \notin \Gamma. \end{cases}$$

Сума одиниць i-го стовпця матриці інцидентності дорівнює степеню відповідної вершини v_i.

Матрицею інцидентності орграфу G порядку n та потужністю сигнатури m називається така прямокутна матриця $A(G) = \left[a_{ij}\right]_{m \times n}$, що

$$a_{ij} = \begin{cases} 1, & \text{якщо вершина } v_i \text{ початок дуги } \square_i; \\ -1, & \text{якщо вершина } v_i \text{ кінець дуги } \square_i; \\ 0, & \text{якщо вершина } v_i \text{ не коінцидентна дузі } \square_i. \end{cases}$$

Матриця суміжності неорієнтованого графу завжди симетрична.

Матриця суміжності орієнтованого графу у загальному випадку не симетрична.

15.2. Приклади виконання завдань.

Приклад 15.1. Запишемо матрицю інцидентності та матрицю суміжності неорієнтованого графу *G* (рис. 15.7).

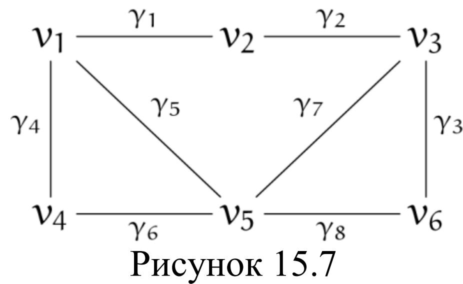


Рисунок 15.7

Матриця інцидентності:

	v_1	v_2	v_3	v_4	v_5	v_6
γ_1	1	1	0	0	0	0
γ_2	0	1	1	0	0	0
γ_3	0	0	1	0	0	1
γ_4	1	0	0	1	0	0
γ_5	1	0	0	0	1	0
γ_6	0	0	0	1	1	0
γ_7	0	0	1	0	1	0
γ_8	0	0	0	0	1	1

Матриця суміжності:

	v_1	v_2	v_3	v_4	v_5	v_6
v_1	0	1	0	1	1	0
v_2	1	0	1	0	0	0
v_3	0	1	0	0	1	1
v_4	1	0	0	0	1	0
v_5	1	0	1	1	0	1
v_6	0	0	1	0	1	0

Приклад 15.2. Запишемо матрицю інцидентності та матрицю суміжності орієнтованого графу *G* (рис. 15.8).

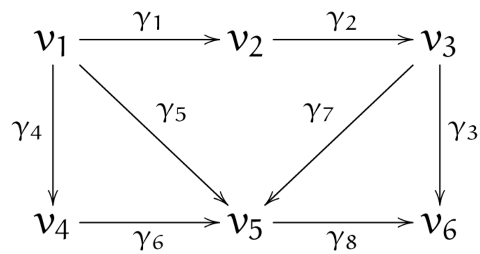


Рисунок 15.8

Матриця інцидентності:

	v_1	v_2	v_3	v_4	v_5	v_6
γ_1	1	–1	0	0	0	0
γ_2	0	1	–1	0	0	0
γ_3	0	0	1	0	0	–1
γ_4	1	0	0	–1	0	0
γ_5	1	0	0	0	–1	0
γ_6	0	0	0	1	–1	0
γ_7	0	0	1	0	–1	0
γ_8	0	0	0	0	1	–1

Матриця суміжності:

	v_1	v_2	v_3	v_4	v_5	v_6
v_1	0	1	0	1	1	0
v_2	0	0	1	0	0	0
v_3	0	0	0	0	1	1
v_4	0	0	0	0	1	0
v_5	0	0	0	0	0	1
v_6	0	0	0	0	0	0

15.4. Завдання для самостійного виконання.

1. Намалювати неорієнтований граф, заданий матрицею суміжності, описати його за допомогою матриці інцидентності та списку ребер.

$$1.\ \begin{pmatrix} 0&0&1&0&0&1\\ 0&1&1&1&1&1\\ 1&0&0&1&0&0\\ 0&0&1&0&0&1\\ 0&0&1&0&0&1\\ 1&0&0&1&1&0 \end{pmatrix} \quad 2.\ \begin{pmatrix} 1&1&1&0&1&1\\ 1&1&0&1&1&0\\ 1&1&0&1&0&0\\ 1&1&0&1&0&0\\ 0&0&0&1&1&1\\ 1&0&1&1&0&0 \end{pmatrix} \quad 3.\ \begin{pmatrix} 0&1&0&1&0&1\\ 0&0&1&1&0&1\\ 1&1&0&0&1&0\\ 0&1&0&0&0&1\\ 1&1&0&0&1&1\\ 0&0&0&0&0&1 \end{pmatrix}$$

$$4.\ \begin{pmatrix} 1&1&1&0&1&1\\ 1&0&0&1&1&0\\ 0&1&0&1&1&0\\ 1&0&1&1&1&1\\ 0&1&1&0&1&1\\ 1&1&1&0&0&1 \end{pmatrix} \quad 5.\ \begin{pmatrix} 0&1&1&0&0&0\\ 1&1&0&0&0&1\\ 0&1&1&0&0&0\\ 0&1&1&0&1&0\\ 0&1&0&1&0&0\\ 0&1&0&0&0&0 \end{pmatrix} \quad 6.\ \begin{pmatrix} 1&0&0&1&0&0\\ 1&1&0&0&1&0\\ 0&0&0&0&1&0\\ 0&0&1&0&0&0\\ 0&1&0&1&0&0\\ 0&1&0&0&0&0 \end{pmatrix},$$

$$7.\ \begin{pmatrix} 0&1&0&1&0&1\\ 1&1&1&1&0&0\\ 0&1&1&0&1&1\\ 1&1&0&1&0&0\\ 0&0&1&0&0&1\\ 1&0&1&0&1&0 \end{pmatrix} \quad 8.\ \begin{pmatrix} 0&1&1&1&1&1\\ 1&1&0&0&1&0\\ 1&0&1&0&0&1\\ 1&0&0&1&1&0\\ 1&1&0&1&0&1\\ 1&0&1&0&1&0 \end{pmatrix} \quad 9.\ \begin{pmatrix} 0&1&0&1&1&1\\ 1&0&1&0&1&1\\ 1&0&0&1&0&1\\ 1&0&1&0&0&1\\ 0&1&0&0&0&0\\ 1&1&0&0&0&0 \end{pmatrix},$$

$$10.\ \begin{pmatrix} 0&0&1&0&1&1\\ 0&1&1&0&0&0\\ 1&1&0&1&0&0\\ 0&1&1&1&0&0\\ 1&0&0&0&0&0\\ 1&1&0&1&0&0 \end{pmatrix} \quad 11.\ \begin{pmatrix} 1&1&1&0&1&1\\ 1&1&0&1&1&0\\ 1&0&0&1&0&0\\ 0&1&1&1&0&0\\ 1&1&0&0&1&1\\ 1&0&0&0&1&0 \end{pmatrix} \quad 12.\ \begin{pmatrix} 1&0&0&1&0&0\\ 0&1&0&0&1&0\\ 0&0&0&0&1&0\\ 1&0&1&0&0&0\\ 0&1&1&0&0&0\\ 0&0&0&0&0&0 \end{pmatrix}.$$

2. Записати матрицю суміжності та матрицю інцидентності для графів (додаток А).

ТЕМА 16.

ОПЕРАЦІЇ ОБ'ЄДНАННЯ, СУМИ ТА ДОБУТКУ ГРАФІВ.

Ключові слова: об'єднання, сума, добуток графів.

16.1. Основні теоретичні відомості.

Для зображення графів у вигляді композиції більш простих графів, існують наступні три операції: об'єднання, суми та добутку.

Об'єднанням графів $G_1 = \langle V_1, \Gamma_1 \rangle$ і $G_2 = \langle V_2, \Gamma_2 \rangle$ називається такий граф $G = \langle V, \Gamma \rangle$ що $V = V_1 \cup V_2$, $\Gamma = \Gamma_1 \cup \Gamma_2$, де $\cup$ – звичайне теоретико-множинне об'єднання.

Сумою графів $G_1 = \langle V_1, \Gamma_1 \rangle$ і $G_2 = \langle V_2, \Gamma_2 \rangle$ називається граф $G = \langle V, \Gamma \rangle$ такий, що $V = V_1 \cup V_2$, $\Gamma = \Gamma_1 \cup \Gamma_2$, причому кожна вершина графу G_1 суміжна з вершинами графу G_2 , і навпаки.

Добутком графів $G_1 = \langle V_1, \Gamma_1 \rangle$ і $G_2 = \langle V_2, \Gamma_2 \rangle$ називається граф $G = \langle V, \Gamma \rangle$, в якому $V = V_1 \times V_2 = \{(v1_i, v2_j) \mid v1_i \in V_1,\ v2_j \in V_2\}$, $B(v1_i, v2_j) = B(v1_i) \times B(v2_j)$, де $B(v1_i, v2_j)$ – окіл вершини $(v1_i, v2_j)$ графу G.

16.2. Приклади виконання завдань.

Приклад 16.1. Нехай задані графи (рис. 16.1).

$G_1 :\ a —— b$, $\quad G_2 :\ c —— d —— e$

Рисунок 16.1

Побудувати суму та добуток графів G_1 та G_2 .

Згідно з означенням сумою графів G_1 та G_2 буде наступний граф:

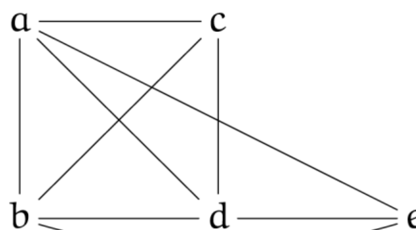


Рисунок 16.2

Необхідні викладки для побудови добутку графів:

$V_1 = \{a, b\}$, $V_2 = \{c, d, e\}$,

$V_1 \times V_2 = \{(a,c), (a,d), (a,e), (b,c), (b,d), (b,e)\}$.

$B((a,c)) = \{b\} \times \{d\} = \{(b,d)\}$,

$B((a,d)) = \{b\} \times \{c,e\} = \{(b,c),(b,e)\}$,

$B((a,e)) = \{b\} \times \{d\} = \{(b,d)\}$,

$B((b,c)) = \{a\} \times \{d\} = \{(a,d)\}$,

$B((b,d)) = \{a\} \times \{c,e\} = \{(a,c),(a,e)\}$,

$B((b,e)) = \{a\} \times \{d\} = \{(a,d)\}$.

Добутком графів G_1 та G_2 буде граф на рис. 16.3.

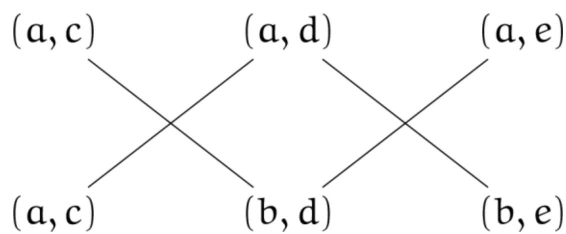


Рисунок 16.3

16.3. Завдання для самостійного виконання.

1. Побудувати суму та добуток графів (рис. 16.4).

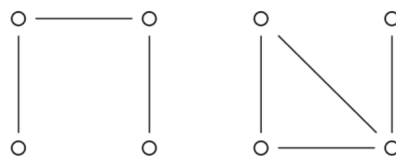


Рисунок 16.4

2. Побудувати суму та добуток двох графів, обираючи пари з представлених на рис. 16.5 графів.

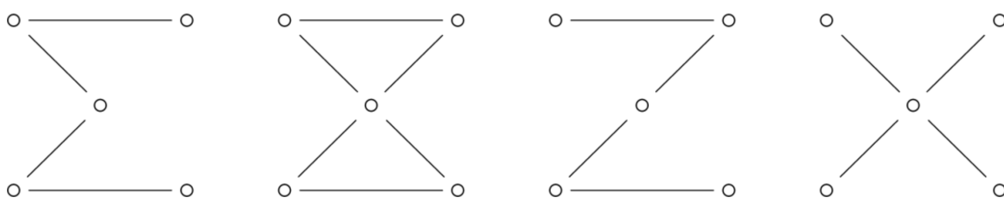


Рисунок 16.5

3. Побудувати суму та добуток графів, заданих матрицями суміжності, обираючи довільні пари.

$$G_1:\begin{matrix} & a & b \\ a & 0 & 1 \\ b & 1 & 0\end{matrix},\quad G_2:\begin{matrix} & 1 & 2 & 3 \\ 1 & 0 & 1 & 0 \\ 2 & 1 & 0 & 1 \\ 3 & 0 & 1 & 0\end{matrix},\quad G_3:\begin{matrix} & a & b & c \\ a & 0 & 1 & 1 \\ b & 1 & 0 & 1 \\ c & 1 & 1 & 0\end{matrix},\quad G_4:\begin{matrix} & c & d & e \\ c & 0 & 1 & 1 \\ d & 1 & 0 & 1 \\ e & 1 & 1 & 0\end{matrix},$$

$$G_5:\begin{matrix} & 1 & 2 & 3 \\ 1 & 0 & 1 & 1 \\ 2 & 1 & 0 & 1 \\ 3 & 1 & 1 & 0\end{matrix},\quad G_6:\begin{matrix} & b & c & d \\ b & 0 & 1 & 1 \\ c & 1 & 0 & 1 \\ d & 1 & 1 & 0\end{matrix},\quad G_7:\begin{matrix} & 1 & 2 \\ 1 & 0 & 1 \\ 2 & 1 & 0\end{matrix},\quad G_8:\begin{matrix} & 3 & 4 & 5 \\ 3 & 0 & 1 & 0 \\ 4 & 1 & 0 & 1 \\ 5 & 0 & 1 & 0\end{matrix}.$$

ТЕМА 17.

МЕТРИЧНІ ВЛАСТИВОСТІ ГРАФІВ.

Ключові слова: центр, радіус, діаметр, медіани графа.

17.1. Основні теоретичні відомості.

Ексцентриситетом вершини графа називається відстань до максимально віддаленої від неї вершини.

Радіус графа – мінімальний ексцентриситет вершин, а *діаметр* – максимальний ексцентриситет вершин.

Центр графа утворюють вершини, у яких ексцентриситет дорівнює радіусу. Центр може мати одну, декілька або всі вершини.

Периферійні вершини мають ексцентриситет, який дорівнює діаметру.

Простий ланцюг з довжиною, яка дорівнює діаметру графа, має назву *діаметральний*.

У зв'язному графі діаметр не більше рангу його матриці суміжності.

Кожне дерево має центр, який складається з однієї або двох суміжних вершин.

Якщо діаметр дерева парний, то дерево має єдиний центр й всі діаметральні ланцюги проходять крізь нього, якщо діаметр непарний, то центрів два й всі діаметральні ланцюги містять ребро, що їх з'єднує.

Нехай $S = \left[s_{ij}\right]_{n\times n}$ – матриця суміжності графа G без петель та $S^k = \{\square_{ij}\}$, де $k \in N$. Тоді β_{ij} дорівнює числу маршрутів довжини k від вершини v_i до вершини v_j .

17.2. Приклади виконання завдань.

Приклад 17.1. Знайти радіус, діаметр та центр графа (рис. 17.1).

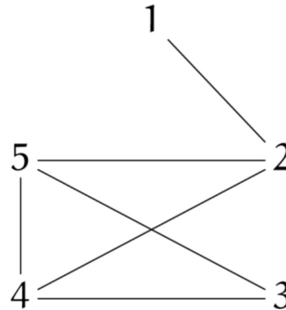


Рисунок 17.1

Знайдемо радіус й діаметр. Знаходячи ланцюги найменшої довжини, обчислимо відстань між вершинами. Результати занесемо в матрицю відстаней. Для неографа ця матриця симетрична. Для скорочення обчислень знаходимо елементи над діагоналлю матриці, заповнюючи другу половину за умовами симетрії:

$$S = \begin{pmatrix} 0 & 1 & 3 & 2 & 2 \\ 1 & 0 & 2 & 1 & 1 \\ 3 & 2 & 0 & 1 & 1 \\ 2 & 1 & 1 & 0 & 1 \\ 2 & 1 & 1 & 1 & 0 \end{pmatrix}.$$

Обчислюємо ексцентриситет кожної вершини. Цю величину можна визначити як максимальний елемент відповідного стовпця (або рядка) матриці відстаней. Отримаємо:

№	1	2	3	4	5
ε	3	2	3	2	2

Радіус графа r – мінімальний ексцентриситет вершин. В даному випадку $r = 2$. Такий ексцентриситет мають вершини 2, 4, 5. Ці вершини утворюють центр графа.

Діаметр графа d – максимальний ексцентриситет вершин. В даному випадку $d = 3$. Такий ексцентриситет мають вершини 1 та 3. Це периферія графа.

Приклад 17.2. За допомогою матриці суміжності знайти радіус, діаметр та центр графа, розглянутого в прикладі 1.

Побудуємо матрицю суміжності графа з прикладу 1.

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{pmatrix}.$$

Будемо заповнювати матрицю відстаней, розглядаючи степені матриці суміжності. В першому наближенні одиниці матриці суміжності показують пари вершин, відстань між якими дорівнює 1 (тобто, вони з’єднані одним ребром):

$$S_{(1)} = \begin{pmatrix} 0 & 1 & - & - & - \\ 1 & 0 & - & 1 & 1 \\ - & - & 0 & 1 & 1 \\ - & 1 & 1 & 0 & 1 \\ - & 1 & 1 & 1 & 0 \end{pmatrix}.$$

Діагональні елементи матриці відстаней – нулі.

Далі знаходимо квадрат матриці суміжності.

$$A^2 = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 3 & 2 & 1 & 1 \\ 0 & 2 & 2 & 1 & 1 \\ 1 & 1 & 1 & 3 & 2 \\ 1 & 1 & 1 & 2 & 3 \end{pmatrix}.$$

Згідно з твердженням про число маршрутів між вершинами 1 та 4, 2 і 3 й так далі, є деяке число маршрутів довжиною 2 (оскільки степінь матриці 2). Важливий лише факт існування маршруту й його довжина, на що вказує ненульовий елемент степені матриці, який не співпадає з елементом, поміченим при обчисленні маршруту меншої довжини. Підставляємо 2 в незаповнені елементи матриці відстаней й отримаємо наступне наближення:

$$S_{(2)} = \begin{pmatrix} 0 & 1 & - & 2 & 2 \\ 1 & 0 & 2 & 1 & 1 \\ - & 2 & 0 & 1 & 1 \\ 2 & 1 & 1 & 0 & 1 \\ 2 & 1 & 1 & 1 & 0 \end{pmatrix}.$$

Залишилося невідомим відстань між вершинами 1 та 3. Будемо помножувати матрицю суміжності A саму на себе то тих пір, поки в матриці A^k не з'явиться ненульовий елемент a_{13}. Тоді відповідний елемент матриці відстаней (або її наближення) буде дорівнювати степені матриці суміжності $s_{13} = k$.

Отримаємо

$$A^3 = \begin{pmatrix} 0 & 3 & 2 & 1 & 1 \\ 3 & 2 & 2 & 6 & 6 \\ 2 & 2 & 2 & 5 & 5 \\ 1 & 6 & 5 & 4 & 5 \\ 1 & 6 & 5 & 5 & 4 \end{pmatrix}$$

Отже, $s_{13} = 3$.

$$S_{(3)} = \begin{pmatrix} 0 & 1 & 3 & 2 & 2 \\ 1 & 0 & 2 & 1 & 1 \\ 3 & 2 & 0 & 1 & 1 \\ 2 & 1 & 1 & 0 & 1 \\ 2 & 1 & 1 & 1 & 0 \end{pmatrix}.$$

Отримана матриця повністю співпадає з матрицею відстаней, знайденою в прикладі 1.

17.3. Завдання для самостійного виконання.

1. Знайти радіус, діаметр, центр графа, який заданий матрицею суміжності:

$$1)\ \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}; \quad 2)\ \begin{pmatrix} 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \end{pmatrix};$$

$$3)\ \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}; \quad 4)\ \begin{pmatrix} 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \end{pmatrix};$$

$$5)\begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}; \qquad 6)\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}.$$

2. Для заданих неографів (додаток А) знайти радіус, діаметр та центр графа.

ТЕМА 18.

ВИЗНАЧЕННЯ ІЗОЛЬОВАНИХ, ТУПИКОВИХ, ВИСЯЧИХ ВЕРШИН. ВИЗНАЧЕННЯ ПЕТЕЛЬ ТА КОНТУРІВ У ГРАФАХ.

Ключові слова: ізольована, тупикова, висяча вершина, петля, контур.

18.1. Основні теоретичні відомості.

Вершина графа називається *ізольованою*, якщо в графі не існує дуг, які інцидентні цій вершині.

Вершина графа називається *висячою*, якщо для всіх дуг графа, які інцидентні цій вершині, вона є початковою.

Вершина графа називається *тупиковою*, якщо для всіх дуг графа, які інцидентні цій вершині, вона є кінцевою.

Виявити всі ізольовані, висячі та тупикові вершини деякого графу G можна за допомогою матриці суміжності графу $S(G)$, а саме сум елементів матриці по рядкам та стовпцях:

$$s^{(j)} = \sum_{i=1}^{n} s_{ij}, \quad j = 1, \ldots, n.$$

$$s_i = \sum_{j=1}^{n} s_{ij}, \; i = 1, \ldots, n.$$

Якщо для деякій k-й вершини графа обидві величини $s^{(k)}$ та s_k, $k \in [1, \ldots, n]$ дорівнюють нулю, то ця вершина буде ізольованою. Якщо тільки величина $s^{(k)} = 0$, то вершина буде висячою. Якщо тільки $s_k = 0$, то вершина буде тупиковою.

Тому, розглядаючи матрицю суміжності вершин графа, знаходять ізольовані вершини, для яких відповідні їм рядки й стовпці матриця мають тільки нульові елементи, висячі вершини, для яких тільки відповідні їм стовпці не мають ненульових елементів, й тупикові вершини, для яких тільки відповідні їм рядки не мають ненульових елементів.

Наявність петель та контурів теж можна виявити за допомогою матриці суміжності. Якщо матриця $S_{[n]}$ суміжності вершин графа має ненульові елементі, які розташовані на головній діагоналі, тобто існують $s_{ii} \neq 0$ при $i \in [1, \ldots, n]$, то вершини графа, які відповідають цим елементам, мають петлі. Рівність нулю усіх елементів матриці суміжності, розташованих на головній діагоналі, відповідає про відсутність петель в графі.

Наявність контурів в графі визначається наступним чином.

Розглянемо матрицю $S_{[n]}$ суміжності вершин графа. Якщо в ній є не рівні нулю елементи головної діагоналі, то дорівнюємо їх до нуля. Отриману матрицю позначимо $Q_{[n]}$ й возведемо в квадрат. При відсутності ненульових елементів головної діагоналі в матриці суміжності в якості матриці $Q_{[n]}$ обираємо її саму.

Елементи матриці $Q^{(2)}_{[n]} = Q^2_{[n]}$ визначають за формулою

$$q^{(2)}_{ij} = \sum_{p=1}^{n} q_{ip} q_{pj}, \ i = 1, \ldots, n, \ j = 1, \ldots, n.$$

Кожен доданок в цьому виразі не дорівнює нулю тільки в тому випадку, коли обидва множника q_{ip} та q_{pj} не рівні нулю. А це можливо тільки тоді, коли існує шлях з вершини i у вершину j, який має дві дуги й проходить через вершину p. Таким чином, значення елементу $q^{(2)}_{ij}$ матриці $Q^{(2)}_{[n]}$ дорівнює кількості шляхів довжини 2, які ведуть з вершини i у вершину j. Отже, матриця $Q^{(2)}_{[n]}$ визначає кількість всіх шляхів довжини 2 в графі.

Не рівні нулю елементи головної діагоналі матриці $Q_{[n]}^{(2)}$, якщо вони є, говорять про наявність контурів в графі довжиною 2.

Для виявлення контурів довжини 3 в матриці $Q_{[n]}^{(2)}$ дорівнюють нулю ненульові елементи головної діагоналі, якщо вони є, й помножують отриману матрицю на матрицю Q_n:

$$Q_{[n]}^{(3)} = Q_{[n]}^{(2)'} Q_{[n]},$$

де $Q_{[n]}^{(2)'}$ − матриця, яка була отримана з матриці $Q_{[n]}^{(2)}$ після порівнювання нулю ненульових елементів головної діагоналі.

Не рівні нулю елементи головної діагоналі матриці $Q_{[n]}^{(3)}$ говорять про наявність контурів довжини 3 в графі. Якщо всі елементи головної діагоналі матриці дорівнюють нулю, то контурів довжини 3 в графі немає.

Матриця $Q_{[n]}^{(k)}$ визначає кількість всіх шляхів довжини k в графі. Так як в графі з n вершинами самий довгий елементарний контур не може мати довжину більш ніж n, то й кількість операцій не перевищує n.

18.2. Приклади виконання завдань.

Приклад 18.1. Визначити наявність ізольованих, висячих, тупикових вершин, також петель та контурів в графі на рис. 18.1.

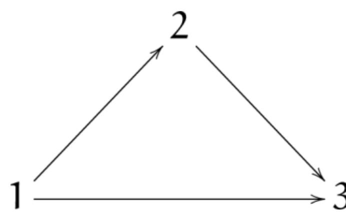


Рисунок 18.1

Матриця суміжності вершин графа має вид:

$$S_3 = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}.$$

В матриці перший стовбець й третій рядок мають тільки нулі, значить, перша вершина в графі – висяча, а третя – тупикова.

Ізольованих вершин нема.

Так як головна діагональ матриці має тільки нулі, то петель в графі немає.

Возведемо матрицю суміжності в квадрат:

$$S_{[3]}^{(2)} = S_{[3]}^{2} = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

Отримана матриця має тільки нулі на головній діагоналі, тому контурів довжини 2 в графі немає.

Возведемо матрицю суміжності в куб.

$$S_{[3]}^{(3)} = S_{[3]}^{3} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

Оскільки отримали нульову матрицю, то контурів в графі немає.

Приклад 18.2. Визначити наявність ізольованих, висячих, тупикових вершин, також петель та контурів в графі на рис. 18.2.

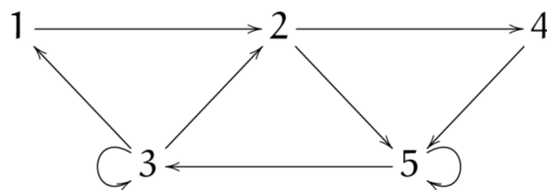


Рисунок 18.2

Матриця суміжності вершин графа має наступний вид:

$$S_{[5]} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}.$$

На головній діагоналі матриці суміжності розташовані два ненульових елементи: $s_{33}=1$, $s_{55}=1$. Значить третя й п’ята вершини графа мають петлі. Ненульових стовпців й рядків в матриці немає, отже, граф не має висячих, ізольованих й тупикових вершин.

Отримав матрицю $Q_{[5]}$ шляхом прирівнювання нулю елементів s_{33} й s_{55}, возведемо її в квадрат.

$$Q_{[5]}^{(2)} = Q_{[5]}^{2} = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{pmatrix}.$$

На головній діагоналі отриманої матриці відсутні ненульові елементи. Отже контурів довжини 2 в графі немає.

Обчислюємо $Q_{[5]}^{(3)}$.

$$Q_{[5]}^{3} = \begin{pmatrix} 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 2 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 \end{pmatrix}.$$

Оскільки на головній діагоналі отриманої матриці стоять ненульові елементи $q_{22}^{(3)}=1$, $q_{33}^{(3)}=1$, $q_{55}^{(3)}=1$, то в графі є контур довжини 3, який походить крізь друг, третю й п’яту вершини графа.

Обчислюємо матрицю $Q_{[5]}^{(4)}$.

$$Q_{[5]}^{4} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 2 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 2 \end{pmatrix}.$$

Аналіз елементів головній діагоналі показує, що в графі є два контури довжиною 4, які проходять через першу, другу, третю й п’яту вершини та другу, третю, четверту й п’яту вершини.

Обчислюємо матрицю $Q_{[5]}^{(5)}$.

$$Q_{[5]}^{5} = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Аналіз елементів головної діагоналі показує, що є один контур довжиною 5, який проходить через всі вершини. Інших контурів, окрім виявлених в графі немає, так як максимальна довжина елементарного контура в графі з п’ятьма вершинами не може бути більш п’яти.

18.3. Завдання для самостійного виконання.

Визначити наявність ізольованих, висячих, тупикових вершин, також петель та контурів в графах додатку Б.

ТЕМА 19.

ЦИКЛОМАТИКА.

Ключові слова: дерево, остов графу, хорда остова, цикломатичне число, цикломатична матриця, векторний простір циклів графу, базис циклів, розріз (коцикл) графу, коцикломатичний ранг, мост.

19.1. Основні теоретичні відомості.

Зв'язний граф, який не містить циклів, називається *деревом*.

Остовом D_G зв'язного графу G називається такий його частковий граф, який є деревом. Усяке ребро графу G, яке не належить остову D_G, називається *хордою* остову.

Твердження 19.1. Будь-який частковий граф, який складається з хорди та остову, має точно один цикл.

Цикломатичне число $\nu(G)$ графу *G* дорівнює числу хорд будь-якого остову в *G*.

Твердження 19.2. Якщо зв'язаний граф *G* містить *n* вершин та *m* ребер, то $\nu(G) = m - n + 1$.

Коцикломатичний ранг $\chi(G)$ графу *G* (ранг розрізів) визначається як кількість ребер в остові D_G цього графу. Отже, $\chi(G) = m - \nu$.

Нехай граф містить *m* ребер та *s* простих циклів. *Цикломатична матриця* графу *G* – це така матриця *C*(*G*) = $[c_{ij}]_{s\times m}$, у якій стовпцям зіставляються ребра графу, а рядкам – прості цикли, причому

$$c_{ij} = \begin{cases} 1, \text{якщо } j-\text{е ребро} \quad \text{входить в } i-\text{й цикл}, \\ 0, \text{у протилежно му випадку}. \end{cases}$$

Кожному простому циклу R_i відповідає вектор-рядок з нулів та одиниць довжини *m*.

Базис циклів графу *G* – це сукупність таких простих циклів R_i графу *G*, що будь-який цикл графу зображується їх лінійною комбінацією за модулем два.

Твердження 19.3. Число базисних циклів графу стале і дорівнює його цикломатичному числу.

Базисний цикл містить тільки одну хорду.

Поділяючою множиною зв'язного графу *G* називається така множина його ребер, вилучення яких робить граф незв'язним.

Розрізом (або, *коциклом*) графу називається така поділяюча множина, яка не містить власної поділяючої підмножини.

Нехай граф містить *m* ребер та *t* простих циклів. *Матриця розрізів* графу *G* – це така матриця *K*(*G*) = $[k_{ij}]_{t\times m}$, у якій стовпцям зіставляються ребра графу, а рядкам – розрізи, причому

$$k_{ij} = \begin{cases} 1, \text{якщо } j-\text{е ребро} \quad \text{входить в } i-\text{й} \quad \text{розріз}, \\ 0, \text{у протилежно му випадку}. \end{cases}$$

Розріз, який складається з одного ребра, називають *мостом*.

19.2. Приклади виконання завдань.

Приклад 19.1. Визначити цикломатичну матрицю та матрицю розрізів графу (рис. 19.1).

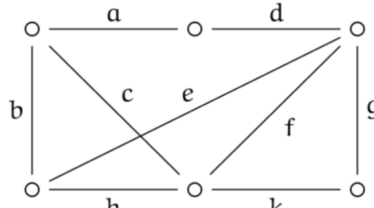


Рисунок 19.1

Граф G містить $n = 6$ вершин та $m = 9$ ребер: $a, b, c, d, e, f, g, h, k$.

Отже, цикломатичне число графу, згідно формулі $\nu(G) = m - n + 1$, дорівнює 4. Згідно твердженню 19.3, граф містить 4 базисних цикла, та, відповідно, чотири хорди. Хордами можуть бути ребра a, b, e, k.

Базисні цикли:

$$R_1 = \{a, c, d, f\},\ R_2 = \{b, c, h\},\ R_3 = \{e, f, h\},\ R_4 = \{f, g, k\}.$$

Запишемо цикломатичну матрицю графу $C(G)$:

	a	b	c	d	e	f	g	h	k
R_1	1	0	1	1	0	1	0	0	0
R_2	0	1	1	0	0	0	0	1	0
R_3	0	0	0	0	1	1	0	1	0
R_4	0	0	0	0	0	1	1	0	1

Усі інші цикли графу G можуть бути одержані з базисних циклів, якщо їх додавати за модулем 2. Таку операцію треба виконати $2^{\nu} - \nu - 1$ раз, де ν – цикломатичне число графу.

Зображаємо остов графу D_G (рис. 19.2).

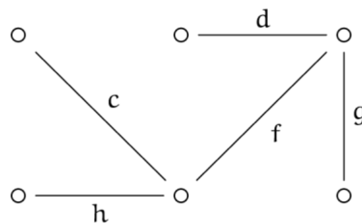


Рисунок 19.2

Базисна система розрізів наступна:

$K_1 = \{a, b, c\}$, $K_2 = \{a, d\}$, $K_3 = \{a, e, f, k\}$, $K_4 = \{g, k\}$, $K_5 = \{b, e, h\}$.

Ця система одержана наступним чином. Вилучається ребро остова. Множина вершин при цьому розпадається на дві не перетинаючи підмножини V_1 і V_2.

Множина всіх ребер в *G*, кожне з яких поєднує вершину з V_1 з вершиною із V_2, а також вилучене ребро остову являється розрізом графа *G*.

Зокрема, розріз $K_3 = \{a, e, f, k\}$ одержано вилученням з остову ребра *f* (рис. 19.3).

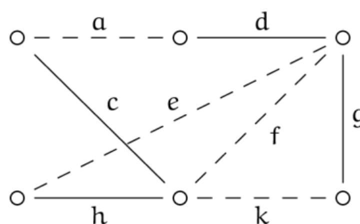


Рисунок 19.3

Запишемо матрицю розрізів графу *K*(*G*) :

	a	*b*	*c*	*d*	*e*	*f*	*g*	*h*	*k*
K_1	1	1	1	0	0	0	0	0	0
K_2	1	0	0	1	0	0	0	0	0
K_3	1	0	0	0	1	1	0	0	1
K_4	0	0	0	0	0	0	1	0	1
K_5	0	1	0	0	1	0	0	1	0

Усі інші розрізи графу *G* можуть бути одержані з базисних розрізів, якщо їх додавати за модулем 2. Таку операцію треба виконати $2^{\chi} - \chi - 1$ раз, де $\square(G)=m-\square$ – коцикломатичний ранг графу *G*.

19.3. Завдання для самостійного виконання.

1. Визначити цикломатичну матрицю та матрицю розрізів графу (додаток А).

ТЕМА 20.

СТІЙКІСТЬ, ПОКРИТТЯ, ЩІЛЬНІСТЬ ГРАФІВ.

Ключові слова: вершинне (реберне) число внутрішньої стійкості графу, число вершинного покриття, щільність графу.

20.1. Основні теоретичні відомості.

Множину вершин графу називають *внутрішнє стійкою*, якщо вершини парами не суміжні. Внутрішнє стійка множина вершин називається *порожнім підграфом*, якщо при додаванні хоча б однієї вершини утворюється ребро (або ребра). Порожній підграф, який не містить деяку вершину графу, містить хоча б одну вершину з її околу.

Максимальна потужність порожнього підграфу графу *G* називається *числом внутрішньої стійкості* $\varepsilon_0(G)$ (або *вершинним числом незалежності* графу). Аналогічно визначається реберне число незалежності $\square_1(G)$.

Максимальна кількість парами не суміжних ребер графу *G* називається *реберним числом незалежності* графу.

Якщо ребро інцидентно вершині, то кажуть, що вони *покривають* один одного. Множина вершин, які покривають усі ребра графу, називається *вершинним покриттям* графу *G*. Мінімальна потужність вершинного покриття називається *числом вершинного покриття* графу $\pi_0(G)$.

Множина ребер, які покривають усі вершини графу називається *реберним покриттям* графу. Мінімальна потужність реберного покриття графу *G* називається *числом реберного покриття* графу $\square_1(G)$.

Для будь-якого нетривіального зв'язного графу $G = \langle V, U \rangle$ має місце співвідношення:

$$\varepsilon_0(G)+\pi_0(G)=\varepsilon_1(G)+\pi_1(G)=|V|.$$

Алгоритм побудови дерева усіх порожніх підграфів наступний:

1. Починаємо з кореня майбутнього дерева, який зважимо заданим графом *G*.

2. Зафіксуємо вершину v_0 графу *G* (*з мінімальним степенем*) та вершини її околу Γ_{v_0}. Вказаним вершинам зіставляємо кінці дуг, які виходять з кореня дерева (краще донизу від кореня). Таким чином, одержимо 1-й ярус дерева.

3. Кожну вершину $v□ \in \{v_0, V_0\}$ яруса зважимо G-неоколом $\tilde{G}(v_\alpha)$ вершини v_α.

4. Кожну вершину v_α яруса вважаємо коренем нового дерева.

5. Процедуру повторюємо для кожної вершини яруса доки у кожного кінця побудованих гілок неокіл буде порожньою множиною.

Щоб запобігти повторення порожніх підграфів, використовують наступне правило.

Закон поглинання: якщо у k-у ярусі дерева вершини v_i та v_j не суміжні та піддерево з коренем v_i побудовано і якщо у піддереві з коренем v_j з'являється дуга з вершиною v_i, то відповідна гілка не будується.

При побудові наступного ярусу, у підграфах, які зважують кінці дуг, вибирається вершина з мінімальним степенем. У такому разі, число висячих вершин дерева співпадає з числом порожніх підграфів даного графу.

Щільність графу.

Щільністю $p(G)$ графу $G = \langle V, U \rangle$, називається максимальна потужність носія серед всіх повних підграфів цього графу.

Алгоритм побудови дерева усіх повних під графів наступний:

1. Корінь дерева зважимо заданим графом G.

2. Зафіксуємо вершину v_0 графу G (*з максимальним степенем*) та вершини її неоколу. Вказаним вершинам зіставляємо кінці дуг, які виходять з кореня дерева.

3. Кожну вершину $v□ \in \{ v_0, \tilde{V}_0 \}$ яруса зважимо G-околом $G(v_\alpha)$ вершини v_α.

4. Кожну вершину v_α ярусу вважаємо коренем нового дерева.

5. Процедуру повторюємо для кожної вершини яруса доки у кожного кінця побудованих гілок окіл буде порожньою множиною.

Щоб запобігти повторення повних підграфів, використовують наступний закон.

Закон поглинання. Якщо у k-у ярусі дерева вершини v_i та v_j суміжні та піддерево з коренем v_i побудовано і якщо у піддереві з коренем v_j з'являється дуга з вершиною v_i, то відповідна гілка не будується.

При побудові дерева вибирається вершина з максимальним степенем. Це забезпечує побудову такого дерева, коли число висячих вершин дерева співпадає з числом повних підграфів даного графу.

20.2. Приклади виконання завдань.

Приклад 20.1. Визначити число внутрішньої стійкості та число вершинного покриття заданого графу *G* (рис. 20.1).

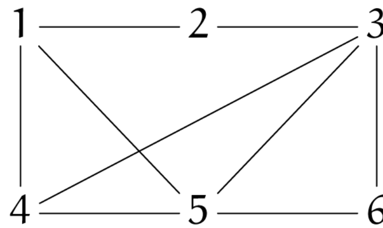


Рисунок 20.1

Число внутрішньої стійкості $\varepsilon_0(G)$ графу *G* є порядок порожнього підграфу графу *G* з максимальною потужністю. Щоб знайти такий підграф, треба побудувати дерево усіх порожніх підграфів графу.

Зауважимо, що при наявності числа внутрішньої стійкості, може бути визначено число вершинного покриття графу:

$$\square_0(G) = |V| - \square_0(G).$$

Вкажемо степені вершин графу.

Вершина	1	2	3	4	5	6
Степінь	3	2	4	3	4	2

Будуємо дерево порожніх підграфів даного графу (рис. 20.2).

Корінь дерева позначаємо порожнім кружком. Вершинами з мінімальним степенем є вершина 2 та вершина 6. Візьмемо вершину 2. Будуємо дугу від кореня, вершину 2 позначаємо подвійним кружком. Вершини 1 та 3 є суміжними до вершини 2, тому від кореня будуємо ще дві дуги.

Кожний кінець побудованого ярусу зважимо неоколом, що приписується на малюнку дерева біля вершин. При відсутності неоколу, біля вершини приписується знак порожньої множини. Вершина 3 з своїм неоколом утинається згідно закону поглинання, бо у даному ярусі є вершина 1, яка не суміжна вершині 3.

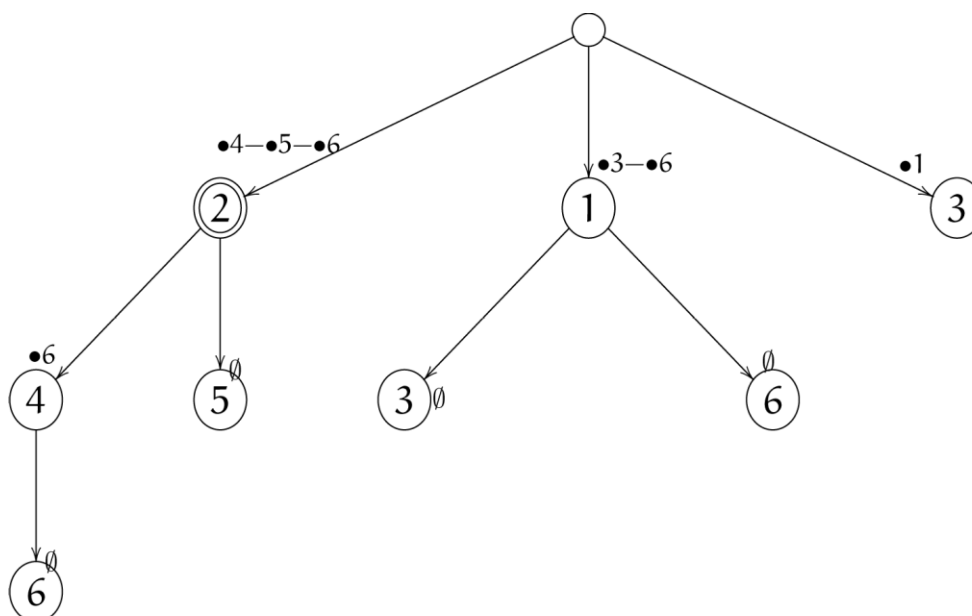


Рисунок 20.2

Таким чином, маємо такі порожні підграфи: E_1 ={2,4,6}, $E_2 = \{2,5\}$, E_3 = {1,3}, E_4 ={1,6}. Легко бачити, що E_{max} = {2, 4, 6} – максимальна кількість парами не суміжних вершин графу *G*, отже, $\varepsilon_0(G) = 3$.

Знаходимо число вершинного покриття $\pi_0(G) = |V| - \varepsilon_0(G)$ = 6–3 = 3.

Більш того, можна вказати саме ті вершини, які покривають усі ребра графа, причому це мінімальна кількість таких вершин. Це вершини $V \setminus E_{max}$ = {1, 3, 5}.

Приклад 20.2. Визначити щільність графу *G* (рис. 20.3).

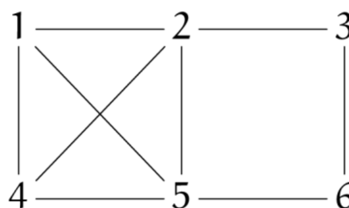


Рисунок 20.3

По-перше, на зображенні даного графу легко вказати сукупності вершин, які можуть бути носієм для повних підграфів. А саме, такими сукупностями є {1, 4, 5}, {3, 4, 5}, {3, 5, 6}. Але, більш зручніше визначити усі повні підграфи можна за допомогою алгоритму.

Будуємо дерево усіх повних підграфів даного графу (рис. 20.4). Корінь дерева позначаємо порожнім кружком. Вкажемо степені вершин графу.

Вершина	1	2	3	4	5	6
Степінь	3	4	2	3	4	2

Отже, маємо дві вершини з максимальним степенем – це вершина 2 та вершина 5. Візьмемо вершину 5. Будуємо дугу від кореня, вершину 5 позначаємо подвійним кружком. Вершина 3 є несуміжною до вершини 5, тому будуємо ще одну дугу з кореня. Кожний кінець побудованого ярусу зважимо околом відповідної вершини. При відсутності околу, біля вершини приписується знак порожньої множини.

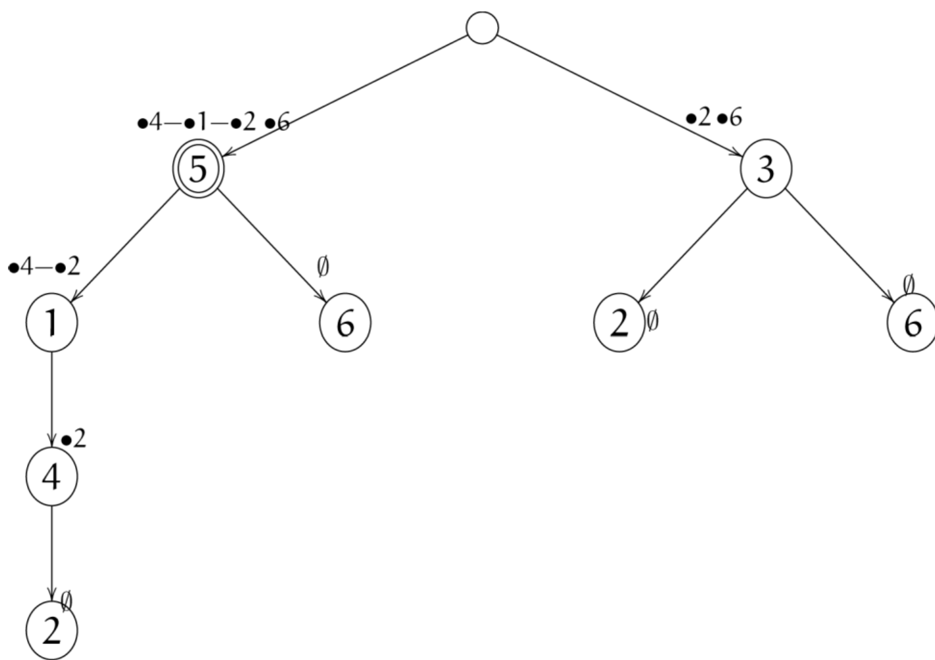


Рисунок 20.4

Таким чином, маємо повний підграф: $E = \{5, 1, 4, 2\}$.

Потужність носія повного підграфу дорівнює 4. Отже, $p(G) = 4$.

20.3. Завдання для самостійного виконання.

1. Для графів додатку А визначити число внутрішньої стійкості та число вершинного покриття.
2. Визначити щільність графів, наведених в додатку А.

ТЕМА 21.

ВИЗНАЧЕННЯ ВЕРШИННИХ ЧИСЕЛ ЗОВНІШНЬОЇ СТІЙКОСТІ ГРАФУ.

Ключові слова: вершинне число зовнішньої стійкості графу.

21.1. Основні теоретичні відомості.

Будемо вважати, що будь-яка вершина графу покриває сама себе, і дві суміжні вершини покривають одна одну. Мінімальна потужність множини вершин, які покривають усі вершини графу *G*, називається *вершинним числом зовнішньої стійкості* графу $\beta_0(G)$.

Будемо вважати, що кожне ребро покриває само себе, і два суміжних ребра покривають одне одного. Мінімальна потужність множини ребер, які покривають усі ребра графу *G*, називається *реберним числом зовнішньої стійкості* графу $\beta_1(G)$.

Визначення зовнішньої стійкості графу.

Модифікованою матрицею суміжності $\tilde{S}(G)$ графу *G* називається диз'юнкція матриці суміжності *S*(*G*) та одиничної матриці $I = diag\{1,1, \ldots, 1\}$.

Визначення числа зовнішньої стійкості неорієнтованого графу зводиться до вилучення мінімального покриття рядків стовпцями модифікованої матриці суміжності цього графу.

Для неорієнтованих графів, завдяки симетричності модифікованої матриці суміжності, покриття рядків стовпцями рівносильно покриттю стовпців рядками.

Для орієнтованих графів розрізняють додатне та від'ємне вершинні числа зовнішньої стійкості (модифікована матриця суміжності орграфу не симетрична).

Додатнім вершинним числом зовнішньої стійкості $\beta_0^+(G)$ орграфу $G = \langle V, \Gamma \rangle$ називається мінімальна потужність множини вершин $V^+ = \{v_i^+\}$ таких, що $\{v_i^+\} \cup \{B(v_i^+)\} = V$.

Від'ємним вершинним числом зовнішньої стійкості $\beta_0^-(G)$ орграфу $G = \langle V, \Gamma \rangle$ називається мінімальна потужність множини вершин $V^- = \{v_i^-\}$ таких, що $\{v_i^-\} \cup \{B^{-1}(v_i^-)\} = V$.

Згідно означенню, число $\beta_0^+(G)$ визначає мінімальну кількість вершин, з яких «спостерігаються» (за один крок) усі вершини графу, а число $\beta_0^-(G)$ визначає мінімальну кількість вершин, які «спостерігаються» усіма вершинами графу.

Очевидно, що число $\beta_0^+(G)$ обчислюється як мінімальне покриття стовпців рядками матриці $\tilde{S}(G)$, а число $\beta_0^-(G)$ – навпаки, як мінімальне покриття рядків стовпцями.

21.2. Приклади виконання завдань.

Приклад 21.1. Визначити число зовнішньої стійкості орграфу *G* (рис. 21.1).

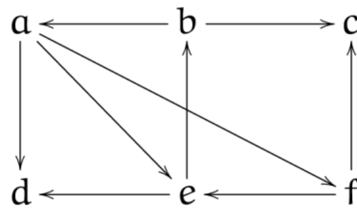


Рисунок 21.1

Обчислимо числа $\beta_0^+(G)$ та $\beta_0^-(G)$ орграфа *G*.

Модифікована матриця суміжності $\tilde{S}(G)$ орграфа *G* має вид:

	a	*b*	*c*	*d*	*e*	*f*
a	1	0	0	1	1	1
b	1	1	1	0	0	0
c	0	0	1	0	0	0
d	0	0	0	1	0	0
e	0	1	0	1	1	0
f	0	0	1	0	1	1

Запишемо покриття стовпців матриці її рядками:

$$(a + b)(b + e)(b + c + f)(a + d + e)(a + e + f)(a + f) =$$
$$= (a + b)(a + f)(b + e)(b + c + f)(a + d + e) =$$
$$= (a + bf)(a + d + e)(b + ce + ef) = (a + bdf + bef)(b + ce + ef) =$$
$$= ab + ace + aef + bdf + bcdef + bdef + bef + bce + bef =$$
$$= ab + ace + aef + bdf + bef.$$

Отже, $\beta_0^+(G)$ = 2, причому на графі G досить чітко видно, що саме з вершин a і b спостерігаються усі вершини графу.

Запишемо покриття рядків матриці її стовпцями:

$$(a + d + e + f)(a + b + c)(c + d)(b + d + e)(c + e + f) = cd.$$

Отже, $\beta_0^-(G)$ = 2. На графі G видно, що саме вершини c і d спостерігаються усіма вершинами графу. Дійсно, вершина c спостерігається вершинами b, f і сама собою, вершина d спостерігається вершинами a, e і сама собою.

При обчисленні покрить користуються законами поглинання та законом дистрибутивності: $x(x + y) = x$, $x + xy = x$, $(x + y)(x + z) = x + yz$.

21.3. Завдання для самостійного виконання.

1. Визначити число зовнішньої стійкості неорграфу (додаток А).
2. Визначити число зовнішньої стійкості орграфу (додаток Б).

ТЕМА 22.

ВИЗНАЧЕННЯ МАКСИМАЛЬНОГО ПОТОКУ ЧЕРЕЗ СІТЬ.

Ключові слова: сіть, граф досяжності, максимальний потік.

22.1. Основні теоретичні відомості.

Сіть – це орієнтований граф $G = \langle V, \Gamma \rangle$ в якому можна виділити дві множини полюсних вершин V^+ і V^- таких, що із кожної вершини $v_i^+ \in V^+$ дуги тільки виходять, а в кожну вершину $v_i^- \in V^-$ дуги тільки заходять, і кожна вершина $v_i \in V \setminus \left(V^+ \cup V^-\right)$ коінцидентна дугам, які входять і які виходять. Крім того, на множині дуг графу задана цілочислова невід'ємна функція $C : \Gamma \rightarrow N \cup 0$, яка визначає «пропускну спроможність» кожної дуги.

Нехай Γ_v^- – множина дуг, які заходять у вершину $v \in V$, Γ_v^+ – множина дуг, які виходять з вершини v. Цілочислова невід'ємна функція φ, яка задана на

множині дуг графа G називається *течією*, якщо вона задовольняє наступним властивостям:

1. $\sum_{\gamma \in \Gamma_v^-} \varphi(\gamma) = \sum_{\gamma \in \Gamma_v^+} \varphi(\gamma)$, де $v \notin V^+ \cup V^-$.
2. $\varphi(\gamma) \geq C(\gamma)$.

Максимальна течія через сіть дорівнює мінімальній пропускній спроможності її розрізу.

Графом досяжності G_g орграфу G називається неорієнтований граф, кожна вершина якого взаємно однозначно відповідає дузі орграфу G і дві вершини поєднані ребром тоді і тільки тоді, коли відповідні їм дуги входять до якогось шляху початкового орграфу G.

Порожній підграф E_i графу досяжності G_g взаємно однозначно визначає розріз сіті G.

Порожній підграф з мінімальною сумою пропускних спроможностей, дорівнює шуканій максимальній течії.

У випадку, коли $C(\gamma) = \varphi(\gamma)$, пропускна спроможність усіх розрізів співпадає.

Алгоритм визначення максимальної течії через сіть:

1. Будуємо граф досяжності G_g сіті G.

2. Визначаємо усі порожні підграфи E_i графу досяжності G_g, тобто розрізи графу G.

3. Обчислюємо пропускну спроможність усіх розрізів E_i.

4. Максимальна течія $\Phi_{max}(G)$ через сіть G визначається розрізом з мінімальною пропускною спроможністю і дорівнює їй.

22.2. Приклади виконання завдань.

Приклад 22.1. Визначити максимальний потік через сіть G (рис. 22.1, кожна дуга має позначку та вказана її пропускна спроможність).

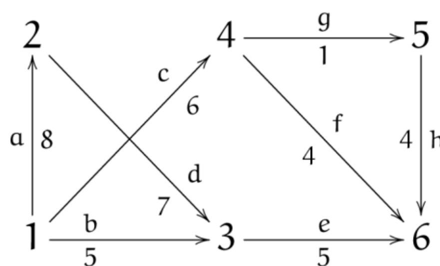


Рисунок 22.1

Будуємо граф досяжності даного графу (рис. 22.2).

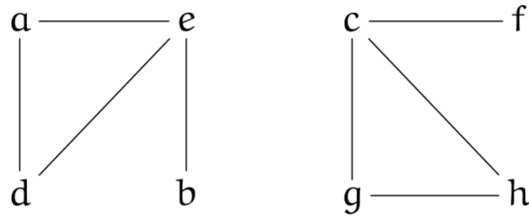


Рисунок 22.2

Як бачимо, граф досяжності складається з двох компонент, які позначимо через G'_g (яка ліворуч) та G''_g (яка праворуч).

Будуємо дерево порожніх підграфів графу досяжності (рис. 22.3).

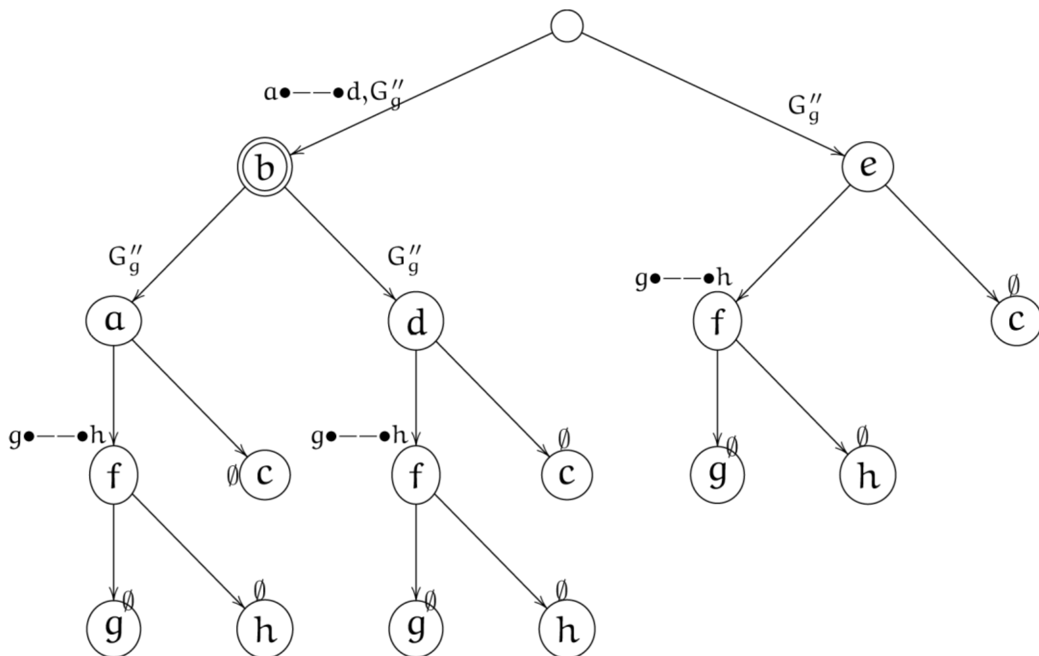


Рисунок 22.3

Обчислимо пропускну спроможність кожного розрізу:

$E_1 = \{b, a, f, g\}$ – 5+8+4+1 = 18, $E_2 = \{b, a, f, h\}$ – 5+8+4+4 = 21,
$E_3 = \{b, a, c\}$ – 5+8+6=19, $E_4 = \{b, d, f, g\}$ – 5+7+4+1 = 17,
$E_5 = \{b, d, f, h\}$ – 5+7+4+4 = 21, $E_6 = \{b, d, c\}$ – 5+7+6 = 18,
$E_7 = \{e, f, g\}$ – 5+4+1=10, $E_8 =\{e, f, h\}$ – 5+4+4 = 13, $E_9 = \{e, c\}$ – 5+6 = 11.

Розріз $\{e, f, g\}$ з мінімальною пропускною спроможністю, яка дорівнює 10, визначає максимальний потік $\Phi_{max} = 10$ через сіть G.

22.3. Завдання для самостійного виконання.

1. Нехай задано сіть *G* (рис. 22.4), пропускна спроможність якої задана в табл. 22.1. Знайти максимальний потік через сіть.

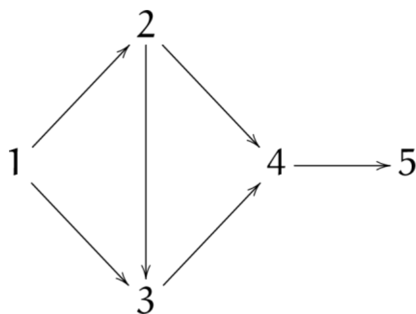


Рисунок 22.4

Таблиця 22.1

Варіант	Дуги					
	(1,2)	(1,3)	(2,3)	(2,4)	(3,4)	(4,5)
1	2	1	3	4	5	3
2	1	2	2	3	1	2
3	2	3	1	5	4	2
4	3	1	2	6	5	2
5	1	3	2	4	5	3

2. Нехай задано сіть *G* (рис. 22.5), пропускна спроможність якої задана в табл. 22.2. Знайти максимальний потік через сіть.

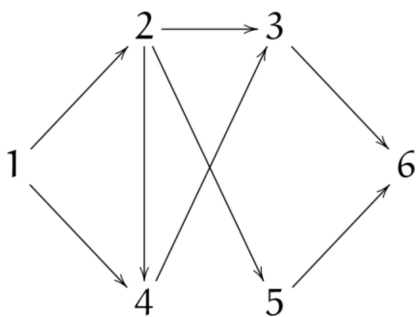


Рисунок 22.5

Таблиця 22.2

1.		2.		3.		4.		5.	
дуга	*C*(γ)	дуга	*C*(γ)	дуга	*C*(γ)	дуга	*C*(γ)	дуга	*C*(γ)
(1,2)	1	(1,2)	2	(1,2)	3	(1,2)	1	(1,2)	3
(1,4)	4	(1,4)	3	(1,4)	2	(1,4)	3	(1,4)	1
(2,3)	2	(2,3)	2	(2,3)	3	(2,3)	1	(2,3)	4
(2,4)	2	(2,4)	3	(2,4)	2	(2,4)	2	(2,4)	3
(2,5)	3	(2,5)	2	(2,5)	1	(2,5)	3	(2,5)	2
(4,3)	3	(4,3)	4	(4,3)	4	(4,3)	2	(4,3)	2
(3,6)	2	(3,6)	2	(3,6)	1	(3,6)	2	(3,6)	1
(5,6)	1	(5,6)	2	(5,6)	1	(5,6)	2	(5,6)	2

3. Нехай задано сіть *G* (рис. 22.6), пропускна спроможність якої задана в табл. 22.3. Знайти максимальний потік через сіть.

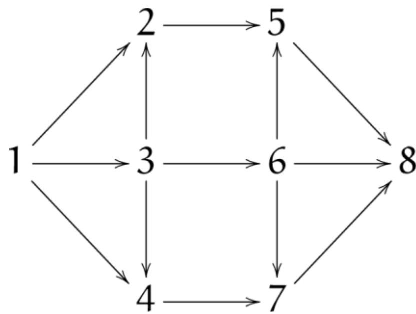


Рисунок 22.6

Таблиця 22.3

1.		2.		3.		4.		5.	
дуга	*C*(γ)	дуга	*C*(γ)	дуга	*C*(γ)	дуга	*C*(γ)	дуга	*C*(γ)
(1,2)	2	(1,2)	1	(1,2)	2	(1,2)	3	(1,2)	1
(1,3)	1	(1,3)	2	(1,3)	3	(1,3)	1	(1,3)	3
(1,4)	3	(1,4)	2	(1,4)	1	(1,4)	2	(1,4)	2
(2,5)	4	(2,5)	3	(2,5)	5	(2,5)	6	(2,5)	4
(3,2)	5	(3,2)	1	(3,2)	4	(3,2)	5	(3,2)	5
(3,4)	3	(3,4)	2	(3,4)	2	(3,4)	2	(3,4)	3
(3,6)	2	(3,6)	4	(3,6)	3	(3,6)	2	(3,6)	3
(4,7)	1	(4,7)	3	(4,7)	2	(4,7)	1	(4,7)	4
(5,8)	3	(5,8)	4	(5,8)	4	(5,8)	3	(5,8)	3
(6,5)	2	(6,5)	2	(6,5)	2	(6,5)	2	(6,5)	2
(6,7)	1	(6,7)	3	(6,7)	1	(6,7)	3	(6,7)	3
(6,8)	1	(6,8)	1	(6,8)	3	(6,8)	1	(6,8)	1
(7,8)	2	(7,8)	1	(7,8)	2	(7,8)	2	(7,8)	1

ТЕМА 23.

РОЗФАРБУВАННЯ ГРАФІВ.

Ключові слова: хроматичне число, хроматичний клас.

23.1. Основні теоретичні відомості.

Розфарбуванням вершин графу $G=\langle V,\Gamma\rangle$ s фарбами називається розбивання носія $V=\bigcup_{k=1}^{s}V_k$ графу G так, що $V_i\cap V_j=\varnothing$ і кожна підмножина V_i не містить ні однієї пари суміжних вершин. Кожній підмножині зіставляється певний колір, в який фарбують усі вершини підмножини.

Хроматичним числом $h(G)$ графу G називається мінімальне число n для якого граф має n-розфарбування вершин.

Розфарбуванням ребер графу $G=\langle V,\Gamma\rangle$, у s фарб називається розбивання сигнатури $\Gamma=\bigcup_{k=1}^{s}\Gamma_k$ графу G так, що $\Gamma_i\cap\Gamma_j=\varnothing$ і кожна підмножина Γ_i не містить ні однієї пари суміжних ребер.

Хроматичним класом $H(G)$ графу G називається мінімальне число n для якого граф має n-розфарбування ребер.

Зрозуміло, що при розфарбуванні вершин (ребер), суміжні вершини (ребра) не будуть одного кольору.

Нехай задано граф $G=\langle V,\Gamma\rangle$. Побудуємо граф $G_\Gamma=\langle X,\Gamma'\rangle$ наступним чином: X співпадає з множиною ребер графа G, і $x'_a\in B(x'_b)$, де $x'_a,x'_b\in\Gamma'$, коли відповідні ребра $a,b\in\Gamma$ графа G суміжні. Тоді, хроматичний клас $H(G)$ графа G співпадає з хроматичним числом графу G_Γ.

Алгоритм мінімального розфарбування вершин (ребер) графу G з використанням властивостей стійкості:

1. Визначаємо усі вершино порожні (реберно порожні) підграфи E_i графу G.

2. Будуємо двовимірну таблицю, кожному рядку якої взаємно однозначно зіставляємо порожній підграф, стовпцю – вершину (ребро) графу G; у клітині (i, j) таблиці записуємо одиницю, якщо j-а вершина (j-е ребро) міститься у i-у порожньому підграфі, в протилежному випадку клітину залишаємо порожньою.

3. Мінімальне покриття стовпців рядками таблиці визначає хроматичне число (хроматичний клас) графу.

У випадку, коли потужність носія графа досить велика і складає декілька десятків і більше вершин, зручніше користуватися наступним алгоритмом розфарбування.

Алгоритм мінімального розфарбування вершин графу *G* з використанням частотних властивостей:

1. Визначаємо пару суміжних вершин v_α і v_β графу, для яких функціонал $\varphi(\{v_\alpha, v_\beta\}) = \frac{|B(v_\alpha) \cap B(v_\beta)|}{|B(v_\alpha) + B(v_\beta)|}$ приймає максимальне значення.

2. Знайдену пару вершин розфарбовуємо та зіставляємо їм стовпці у двовимірній таблиці. Рядкам таблиці зіставляємо не розфарбовані вершини (вони ж заголовки рядків), суміжні вершинам знайденої пари. У клітині (*i*, *j*) таблиці записуємо одиницю, якщо *j*-а та *i*-а вершини суміжні, та 0 у протилежному випадку.

3. Вибираємо рядок з максимальним числом одиниць.

4. Якщо для виділеного рядка знайдеться стовпець, на перетині з яким стоїть 0, то відповідні вершини склеюють, що записують у заголовку відповідного стовпця наступної таблиці. При відсутності нуля, к таблиці додається новий стовпець, якому зіставляється вершина вибраного рядка. Рядки наступної таблиці формуються знову по принципу суміжності.

5. Алгоритм повторюють доки є рядки, тобто доки є хоч би одна не розфарбована вершина.

6. Кількість остаточно одержаних стовпців дорівнює кількості фарб для мінімального розфарбування вершин, а заголовки цих стовпців визначають порожні підграфи.

23.2. Приклади виконання завдань.

Приклад 23.1. Визначити хроматичне число і розфарбувати вершини графу *G* (рис. 23.1) за допомогою алгоритму з використанням порожніх підграфів:

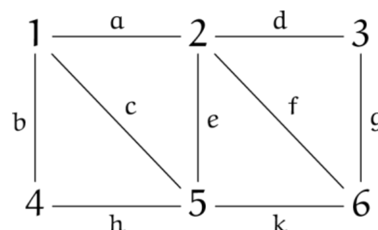


Рисунок 23.1

Побудуємо дерево порожніх підграфів графу *G* (рис. 23.2).

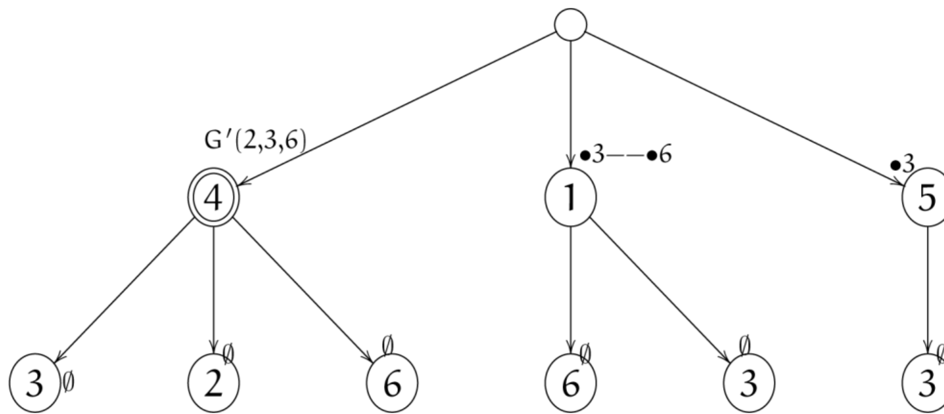


Рисунок 23.2

Побудуємо двовимірну таблицю порожніх графів і вершин:

	1	2	3	4	5	6
$\{4,3\}$			1	1		
√$\{4,2\}$		1		1		
$\{4,6\}$				1		1
√$\{1,6\}$	1					1
$\{1,3\}$	1		1			
√$\{5,3\}$			1		1	

Мінімальне покриття стовпців таблиці її рядками наступне: {4, 2}, {1, 6}, {5, 3}.

Хроматичне число графу $h(G\ ) = 3$. Для розфарбування вершин графу потрібно три фарби, причому підмножина вершин $V_1 = \{4, 2\}$ має один колір, підмножина $V_2 = \{1, 6\}$ – другий, а підмножина $V_3 = \{5, 3\}$ – третій.

Приклад 23.2. Визначити хроматичний клас і розфарбувати ребра графу *G* за допомогою алгоритму з використанням частотних властивостей графу.

Запишемо окіл кожного ребра графу:

$B(a) = \{b, c, d, e, f\}$; $B(b) = \{a, c, h\}$; $B(c) = \{a, b, e, h, k\}$; $B(d) = \{a, e, f, g\}$;
$B(e) = \{a, c, d, f, h, k\}$; $B(f) = \{a, d, e, g, k\}$; $B(g) = \{d, f, k\}$; $B(h) = \{b, c, e, k\}$;
$B(k) = \{c, e, f, g, h\}$.

Обчислимо функціонал для кожної пари суміжних ребер:

$\varphi(a,b)=\frac{1}{8}$, $\varphi(a,c)=\frac{2}{10}$, $\varphi(a,d)=\frac{2}{9}$, $\varphi(a,e)=\frac{3}{11}$, $\varphi(a,f)=\frac{2}{10}$, $\varphi(b,c)=\frac{2}{8}$,
$\varphi(b,h)=\frac{1}{7}$, $\varphi(c,e)=\frac{3}{11}$, $\varphi(c,h)=\frac{1}{3}$, $\varphi(c,k)=\frac{2}{10}$, $\varphi(d,e)=\frac{2}{10}$, $\varphi(d,f)=\frac{1}{3}$,
$\varphi(d,g)=\frac{1}{7}$, $\varphi(e,f)=\frac{3}{11}$, $\varphi(e,h)=\frac{2}{10}$, $\varphi(e,k)=\frac{3}{11}$, $\varphi(f,g)=\frac{2}{8}$, $\varphi(f,k)=\frac{1}{5}$,
$\varphi(g,k)=\frac{1}{8}$.

Виконуючи порівняння, визначаємо, що максимальне значення функціонала дорівнює $\frac{1}{3}$ для ребер *c*, *h* або *d*, *f*. Для визначеності візьмемо ребра *c* і *h*. Будуємо таблицю:

	c	*h*		*c*	*h*	*b*		*c*	*a, h*	*b, e*
a	1	0	*a*	1	0	1	*k*	1	1	1
b	1	1	*e*	1	1	0	*f*	0	1	1
e	1	1	*k*	1	1	0	*d*	0	1	1
k	1	1								

	c, f	*a, h*	*b, e*	*k*		*c, f*	*a, h*	*b, e*	*k, d*
d	1	1	1	0	*g*	1	0	0	1
g	1	0	0	1					

З таблиці бачимо, що ребро *g* може бути склеєно або з ребрами *a*, *h*, або з ребрами *b*, *e*. Хроматичний клас графу *H*(*G*) = 4.

Для розфарбування ребер графу потрібно чотири фарби, причому підмножина ребер $\Gamma_1 = \{f, c\}$ має один колір, підмножина $\Gamma_2 = \{a, g, h\}$ – другий, $\Gamma_3 = \{b, e\}$ – третій, та $\Gamma_4 = \{k, d\}$ – четвертий.

23.3. Завдання для самостійного виконання.

Визначити хроматичне число і розфарбувати вершини графу *G* (додаток А) за допомогою алгоритму з використанням порожніх підграфів та за допомогою алгоритму з використанням частотних властивостей графу.

ТЕМА 24.

АЛГОРИТМІЧНІ ЗАДАЧІ НА ГРАФАХ.

Ключові слова: зважений граф, найкоротший шлях, алгоритм Дейкстри, вага шляху, граф Ейлера.

24.1. Основні теоретичні відомості.

Пошук відстані між вершинами орграфу.

Множину кінців дуг, початком яких є вершина x деякого орграфу $G = \langle V, \Gamma \rangle$, назвемо *образом* цієї вершини і позначимо його $G(x)$. Множину початків дуг, кінцем яких є вершина x, назвемо *прообразом* $G^{-1}(x)$ цієї вершини.

Нехай $G = \langle V, \Gamma \rangle$ орієнтований граф порядку n $(n \geq 2)$.

Візьмемо дві різні вершини графу $v, w \in V$.

Опишемо алгоритм пошуку відстані та відповідного їй мінімального шляху між вершинами v і w:

1. Нехай $B_1(v) = G(v)$, $B_2(v) = G(B_1(v))$, … , $B_k(v) = G(B_{k-1}(v))$. Припустимо, що $r(v,w)=0$, $i = 0$.

2. Якщо $B_{k+i}(v) = \varnothing$ для деякого k, або $w \notin B_{n-1}(v)$, то вершини v і w незв'язні, що завершує алгоритм. В іншому випадку переходимо до наступного кроку.

3. Якщо $w \notin B_{k+i}(v)$, то $r(v,w) = k + i$, і переходимо до кроку 4. В іншому випадку існує шлях із v у w завдовжки $k + i$, причому цей шлях є мінімальним. Послідовність вершин v, w_1, w_2, … ,w_{k+i-1}, w, де

$$w_{k+i-1} \in B_{k+i-1}(v) \cap G^{-1}(w); \quad w_{k+i-2} \in B_{k+i-2}(v) \cap G^{-1}(w_{k+i-1});$$
$$\dots$$
$$w_1 \in B(v) \cap G^{-1}(w_2),$$

є тим самим мінімальним шляхом.

4. Покладаємо $i = i + 1$, визначаємо $B_{k+i}(v)$ і переходимо до кроку 2.

Знаходження довжини найкоротшого шляху.

Зважений граф − граф, ребрам якого приписані деякі числа (ваги).

Завдання про найкоротші шляхи розглядається для зважених орієнтованих графів, дугам яких приписані ненегативні ваги.

Всякому неорієнтованому графові можна зіставити орграф, який виходить з даного заміною кожного неорієнтованого ребра (i, j) парою орієнтованих ребер (i, j) і (j, i) з вагою, рівною вазі початкового неорієнтованого ребра.

Якщо граф не зважений, можна вважати, що всі ребра мають одну і ту ж вагу. Позначимо через w_{ij} вагу дуги (i, j). Неіснуючі дуги вважатимемо дугами з нескінченною вагою.

Вагою (*довжиною*) *шляху* називається сума ваги дуг в цьому шляху.

Задача знаходження довжини найкоротшого шляху, між двома вершинами s (початок) і f (кінець) і відшукання самого шляху може бути вирішена за допомогою алгоритму Дейкстри.

Алгоритм Дейкстри.

В процесі роботи алгоритму вершинам графа привласнюються мітки: тимчасові або постійні.

Спочатку вершині s привласнюється постійна мітка 0 (нульова відстань до самої себе), а кожною з решти вершин приписується тимчасова мітка ∞.

На кожному кроці одній вершині з тимчасовою міткою приписується постійна мітка і пошук продовжується далі. Мітки міняються таким чином:

1) кожній вершині j, що має тимчасову мітку, привласнюється нова тимчасова мітка – найменша з її тимчасової мітки і чисел w_{ij} – остаточна мітка вершини i, де i – вершина, якою привласнена постійна мітка на попередньому кроці.

2) знаходиться найменша зі всіх тимчасових міток, яка стає постійною міткою самої вершини. У разі рівності вибирається будь-яка з них.

Процес продовжується до тих пір, поки вершина f не отримає постійну мітку. Першою вершиною з постійною міткою буде s, яка знаходиться на відстані 0 від себе. Наступною, такою, що отримала постійну мітку, буде вершина, найближча до s. Третьою вершиною, що отримала постійну мітку, буде друга по близькості до s, і т.д. Постійна мітка кожної вершини – це найкоротша відстань до цієї вершини від початкової вершини s.

Сам шлях можна обчислити, якщо використовувати в процесі алгоритму покажчик *pre(v)* на вершину з постійною міткою, за допомогою якої пройшла остання зміна значення мітки для v. Коли мітка вершини v стає постійною, *pre(v)* вказує на вершину, яка передувала вершині v в найкоротшій відстані.

Вершини, які входять в найкоротший шлях, мають наступний вид: *s*,…, *pre*(*pre*(*pre*(*f*))), *pre*(*pre*(*f*)), *pre*(*f*), *f*.

Ейлерові графи.

Цикл називається *ейлеровим*, якщо кожне ребро графу бере участь у його утворенні один раз; *граф* який містить такий цикл називається *ейлеровим*.

Можна сказати, що ейлерові графи – це такі графи, які можна зобразити «одним розчерком пера», тобто, не відриваючи олівець від листа, починаючи і закінчуючи в одній і тій самій точці і проходячи ребра тільки один раз.

Критерій Ейлера. Зв'язний граф $G = \langle V, \Gamma \rangle$ є ейлеровим тоді і тільки тоді, коли степені всіх його вершин парні. Майже усі графи не є ейлеровими.

Алгоритм побудови ейлеревого циклу:

1) вибираємо довільну вершину $a \in V$;

2) будуємо довільний ланцюг *P* з початком у вершині *a*. Оскільки локальні степені всіх вершин графу є парні, то ланцюг може завершитись тільки в *a* (тобто він є циклом);

3) якщо $P(a, a)$ містить не всі ребра графу $G(V)$, то будуємо граф $G_1 = G - P(a, a)$, в якому всі вершини мають теж парний локальний степінь. Оскільки граф $G(V)$ є зв’язним, то серед вершин $P(a, a)$ має знайтись вершина *b*, яка зв’язана ребром хоча б з однією вершиною графу $G(V)$ (інакше граф *G* був би незв’язним);

4) будуємо з ребер графу G_1 ланцюг P', що починається у вершині *b* і може закінчуватись тільки у *b*; з ланцюгів *P* і P' будуємо новий цикл:

$$P_1(a,a) = P(a,b) \cup P(b,b) \cup P(b,a);$$

5) якщо $P_1(a, a)$ не містить всіх ребер графу $G(V)$, то, за аналогією з кроком 3) будуємо граф $G_2 = G - P_1(a, a)$ і т.д.

З огляду на скінченність графу, цей процес зупинитися, коли всі ребра графу $G(V)$ будуть вичерпані.

Узагальнюючи задачу Ейлера можна шукати найменшу кількість ланцюгів P_1, які не перетинаються по ребрах і покривають увесь зв’язний граф $G(V)$.

Нехай $G(V)$ – скінченний зв’язний граф з *k* вершинами непарного локального степеня. Тоді мінімальна кількість ланцюгів, які не перетинаються по ребрах і покривають граф *G*, дорівнює $k/2$.

Алгоритм побудови ланцюгів P_i :

1) з'єднуємо довільним чином пари вершин з непарним локальним степенем (для цього необхідно $k/2$ ребер). При цьому утворюється граф G_1, всі вершини якого мають парний степінь;

2) граф G_1 є ейлеровим і в ньому існує ейлерів цикл S;

3) після відкидання з циклу S доданих на кроці 1) $k/2$ ребер, отримуємо $k/2$ ланцюгів, які покривають весь граф G.

Ейлерів цикл в ейлеревому графові G можна отримати за допомогою *алгоритму Флері*. Він заснований на наступних двох правилах.

1) Виберемо довільну вершину u графа G, і довільному ребру (u, v) припишемо номер 1. Потім викреслимо це ребро і перейдемо у вершину v .

2) Нехай w − вершина, в яку ми прийшли в результаті виконання попереднього кроку, k − номер, привласнений деякому ребру на цьому кроці. Вибираємо будь-яке ребро, інцидентне вершине w, причому перешийок вибираємо тільки у тому випадку, коли немає інших ребер. Вибраному ребру привласнюємо номер k_1 і викреслюємо його.

Алгоритм Флері закінчує роботу, коли всі ребра графа викреслені, тобто занумеровані. Номер, привласнений ребру, указує, яким по рахунку це ребро проходиться в ейлеревому циклі.

24.2. Приклади виконання завдань.

Приклад 24.1. Визначити мінімальний шлях від вершини a до вершини f у орграфі G (рис. 24.1).

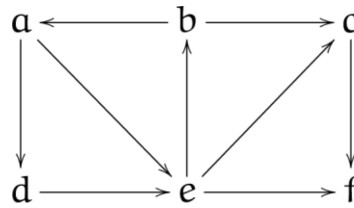


Рисунок 24.1

Запишемо матрицю суміжності $S(G)$ орграфу:

	a	*b*	*c*	*d*	*e*	*f*
a	0	0	0	1	1	0
b	1	0	1	0	0	0
c	0	0	0	0	0	1
d	0	0	0	0	1	0
e	0	1	1	0	0	1
f	0	0	0	0	0	0

Діючи згідно алгоритму, послідовно знаходимо:

$$G(a) = B_1(a) = \{d,e\}.$$

Так як $B_1(d) = \{e\}$ і $B_1(e) = \{b, c, f\}$, то

$$B_2(a) = G(B_1(a)) \setminus \{a, d, e\} = \{b, c, f\}.$$

Отже, $f \in B_2(a)$, $r(a, f) = 2$.

Тепер знайдемо сам мінімальний шлях:

$$B_1(a) \cap G^{-1}(f) = \{d,e\} \cap \{c,e\} = \{e\}.$$

Тоді, *aef* – шуканий мінімальний шлях.

Приклад 24.2. Знайти найменшу кількість ланцюгів, які не перетинаються по ребрах і покривають увесь зв'язний граф (рис. 24.2).

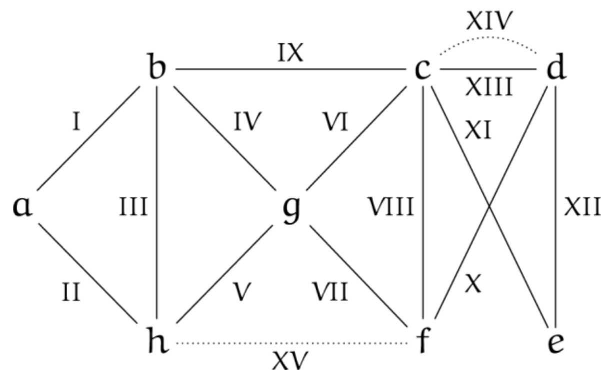


Рисунок 24.2

Локальні степені вершин графу:

Вершина	*a*	*b*	*c*	*d*	*e*	*f*	*g*	*h*
Степінь	2	4	5	3	2	3	4	3

Маємо 4 вершини з непарними степенями: *c, d, f, h*. Отже, $k = 4$. З'єднаємо вершину *c* з вершиною *d* та вершину *f* з вершиною *h* (на рис. 24.2 ці ребра позначені пунктирними лініями). Одержимо граф G_1.

Поетапно побудуємо для утвореного графу G_1 цикл Ейлера:

а) P_1 = (I, III, II);

б) P_2 = (I, IX, VI, IV, III, II);

в) P_3 = (I, IX, XIII, XII, XI, VI, IV, III, II);

г) P_4 = (I, IX, XIV, X, VIII, XIII, XII, XI, VI, IV, III, II);

д) P_5 = (I, IX, XIV, X, VIII, XIII, XII, XI, VI, VII, XV, V, IV, III, II).

Віднімаючи додані раніше ребра XIV і XV, отримаємо три ланцюги:

1) (I, X);

2) (X, VIII, XIII, XII, XI, VI, VII);

3) (V, IV, III, II).

Зауважимо, що перший і третій ланцюги мають спільний кінець – вершину a. Якщо виконати «склейку» цих ланцюгів, отримаємо остаточно:

1) (V, IV, III, II, I, IX);

2) (X, VIII, XIII, XII, XI, VI, VII).

Приклад 24.3. Знайти найкоротшу відстань між вершинами b і g в заданному орієнтованому графі (рис. 24.3).

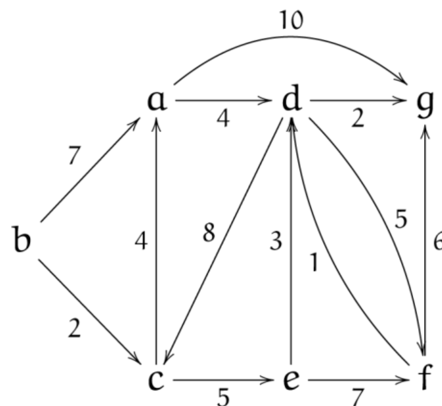


Рисунок 24.3

Запишемо матрицю відстаней графа.

	a	b	c	d	e	f	g
a	0	∞	∞	4	∞	∞	10
b	7	0	2	∞	∞	∞	∞
c	4	∞	0	∞	5	∞	∞
d	∞	∞	8	0	∞	5	2
e	∞	∞	∞	∞	∞	7	∞
f	∞	∞	∞	1	∞	0	6
g	∞	∞	∞	∞	∞	∞	0

Роботу алгоритму по кроках проілюструємо за допомогою табл. 2.1.

Таблиця 24.1

a	*b*	*c*	*d*	*e*	*f*	*g*	
∞	0	∞	∞	∞	∞	∞	Початкова вершина *b* позначена 0.
7		2	∞	∞	∞	∞	Перераховуємо мітки вершин, які пов'язані з вершиною *b* дугою.
7		2	∞	∞	∞	∞	Найменша мітка (мітка вершини *c*) стає постійною.
6			∞	7	∞	∞	Перераховуємо мітки вершин, які пов'язані з вершиною *c* дугою.
6			∞	7	∞	∞	Найменша мітка (мітка вершини *a*) стає постійною.
			10	7	∞	16	Перераховуємо мітки вершин, які пов'язані з вершиною *a* дугою.
			10	7	∞	16	Найменша мітка (мітка вершини *e*) стає постійною.
			10		14	16	Перераховуємо мітки вершин, які пов'язані з вершиною *e* дугою.
			10		14	16	Найменша мітка (мітка вершини *d*) стає постійною.
					14	12	Перераховуємо мітки вершин, які пов'язані з вершиною *d* дугою.
					14	12	Вершина *g* отримує постійну мітку, побудова таблиці закінчується.

Найкоротший шлях від вершини *b* до вершини *g* дорівнює 12.

Знайдемо найкоротший шлях від *b* до *g*. Для цього розглянемо стовбець для *g* знизу догори й знайдемо вершину з постійною міткою, при перерахунку по якій мітка вершини *g* отримала значення 12. Це вершина *d*. Отже, *pre*(*g*) = *d*. Далі, розглядаємо стовбець для вершини *d*. Знаходимо *pre*(*d*) = *a*. Аналогічно, *pre*(*a*) = *c* й *pre*(*c*) = *b*. Таким чином, ми отримали наступний найкоротший шлях від *b* до *g*: $b \to c \to a \to d \to g$. Його довжина дорівнює 2 + 4 + 4 + 2 = 12.

24.3. Завдання для самостійного виконання.

1. Визначити мінімальний шлях для кожної пари вершин орграфу *G* (додаток Б).

2. Нехай граф задано матрицею суміжності. Визначити, чи є граф ейлеровим. В разі позитивної відповіді, вказати в графі ейлерів цикл.

$$1)\ \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix};\quad 2)\ \begin{pmatrix} 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \end{pmatrix};$$

$$3)\ \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

3. Зважений орієнтований граф заданий матрицею відстаней:

	1	2	3	4	5	6	7
1	0	3	∞	∞	2	7	∞
2	∞	0	5	∞	1	∞	∞
3	∞	∞	0	1	∞	∞	∞
4	∞	3	∞	0	∞	2	∞
5	∞	∞	∞	2	0	1	4
6	∞	∞	5	∞	∞	0	9
7	2	∞	∞	5	1	∞	0

Використовуючи алгоритм Дейкстри, по матриці відстаней знайти:

1) найкоротшу відстань й найкоротший шлях між вершинами 1 та 3, між вершинами 3 та 7;

2) найкоротші відстані й найкоротші шляхи від вершин 2 та 5 до решти вершин.

4. Зважений неорієнтований граф заданий матрицею відстаней:

	1	2	3	4	5	6	7
1	0	3	∞	∞	2	7	∞
2	3	0	5	∞	1	∞	8
3	∞	5	0	1	∞	5	∞
4	∞	∞	1	0	2	∞	3
5	2	∞	∞	2	0	1	4
6	7	∞	5	∞	1	0	9
7	∞	8	∞	3	1	9	0

Використовуючи алгоритм Дейкстри, по матриці відстаней знайти:

1) найкоротшу відстань й найкоротший шлях між вершинами 4 та 3, між вершинами 3 та 6;

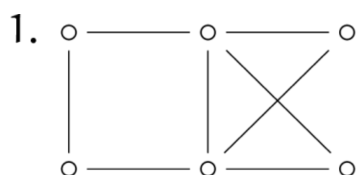
2) найкоротші відстані й найкоротші шляхи від вершин 2 та 5 до решти вершин.

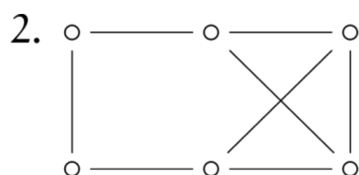
ЛІТЕРАТУРА

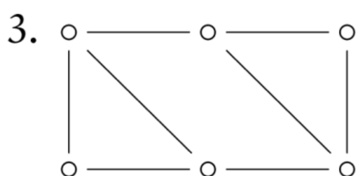
1. Bogart K., Drysdale S., Stein C. Discrete Math for Scince Students. 2004. – 344 p.
2. Kenneth H. Rosen. Discrete Mathematics and Its Applications. Seventh Edition. Monmouth University, Published by McGraw-Hill, 2012. – 1071 p.
3. Lovasz L., Vesztergombi K. Discrete Mathematics. Lecture Notes, Yale University, Spring 1999. – 139 p.
4. Marsel B. Finan. Lecture Notes in Discrete Mathematics. 2001. – 224 p.
5. Азарнова Т. В., Булгакова И. Н. Методические указания для решения задач по курсу Дискретная математика. – Воронеж: Лаб. Опер. полиг. ВГУ, 2000. – 50 с.
6. Акимов О. Е. Дискретная математика: логика, группы, графы. – М.: Лаборатория Базовых Знаний, 2001. – 352 с.
7. Бардачов Ю. М., Соколова Н. А., Ходаков В. Є. Дискретна математика. – Київ: Вища школа, 2007. – 383 с.
8. Бондарчук Ю. В., Олійник Б. В. Основи дискретної математики (для студентів-інформатиків). Національний університет «Києво Могилянська Академія», 2007. – 138 с.
9. Горбатов В. А. Фундаментальные основы дискретной математики. Информационная математика – М.: Наука. Физматлит, 2000. – 544 с.
10. Донской В. И. Дискретная математика. – Симферополь: Сонат, 2000. – 360 с.
11. Жильцова Л. П., Смирнова Т. Г. Основы теории графов и теории кодирования в примерах и задачах: Учебное пособие. – Нижний Новгород: Изд. ННГУ, 2005. – 307 с.
12. Кирсанов М. Н. Графы в Maple. Задачи, алгоритмы, программы. Пособие по дискретной математике для студентов университетов. – Москва: Физматлит, 2007. – 168 с.
13. Сигорский В. П. Математический аппарат инженера. – Киев: Техника, 1977. – 768 с.

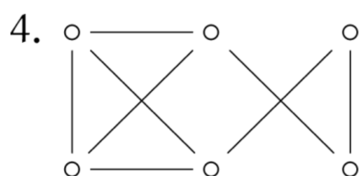
ДОДАТОК А

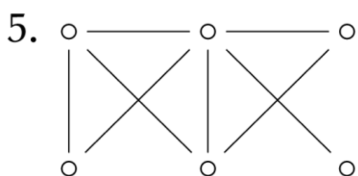
СПИСОК НЕОРІЄНТОВАНИХ ГРАФІВ

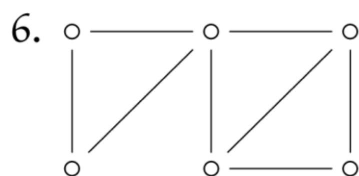


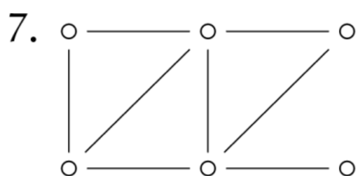


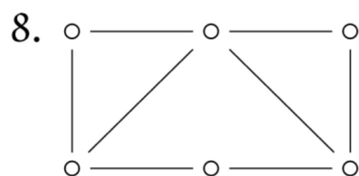


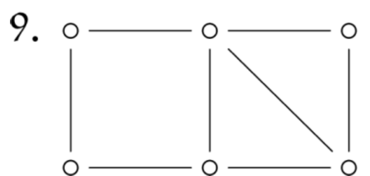


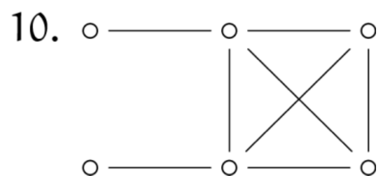






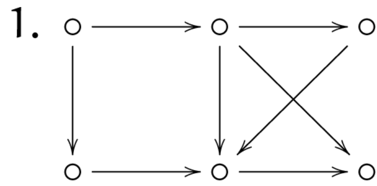


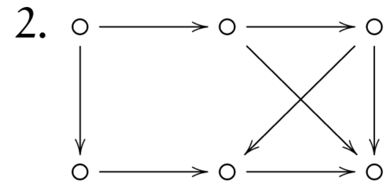


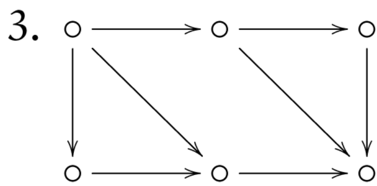


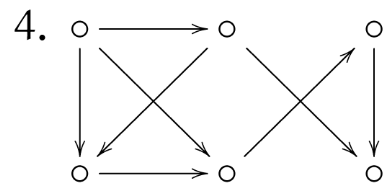
ДОДАТОК Б

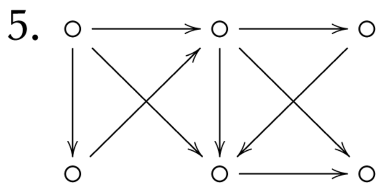
СПИСОК ОРІЄНТОВАНИХ ГРАФІВ

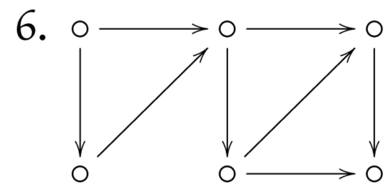


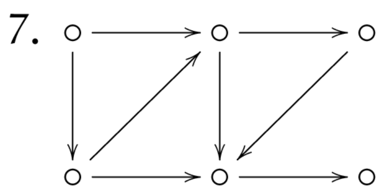


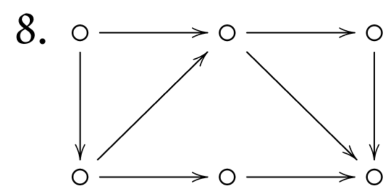


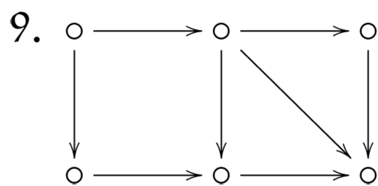


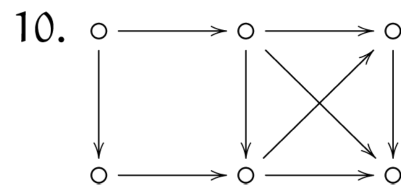












Підп. до друку 25.03.2016р. Формат 60х84/16. Папір офсетний.
Ум.-друк. арк. 9. Наклад 300 прим. Зам. № 20-16.
Ціна договірна

Видавництво та друкарня "Технологічний Центр"
Свідоцтво про внесення суб'єкта видавничої справи ДК № 4452 від 10.12.2012
Адреса: 61145, м. Харків, вул. Шатилова дача, 4
